buildings

MDPI

Article

Exploring Built Environment Visual Interactions: A SoftBIM Data-Driven Approach for a Database About the Outdoor View

Matteo Cavaglià *, Alberto Speroni, Juan Diego Blanco Cadena, Andrea Giovanni Mainini and Tiziana Poli

Department of Architecture, Built Environment and Construction Engineering, Politecnico di Milano, Via Ponzio 31, 20133 Milan, Italy; alberto.speroni@polimi.it (A.S.); juandiego.blanco@polimi.it (J.D.B.C.); andreagiovanni.mainini@polimi.it (A.G.M.); tiziana.poli@polimi.it (T.P.)

* Correspondence: matteo.cavaglia@polimi.it

Abstract: Windows and glazed facades provide outdoor views, serving as vital sources of visual information that aid navigation and interaction within buildings. These views can trigger psychological and physiological responses, affecting individual well-being. However, optimizing outdoor view quality is challenging due to the complex interplay of factors influencing the building's experience of vision. Managing the complexity of optimizing outdoor view quality within current digital frameworks for building design presents significant challenges. A key issue lies in the ambiguity of certain visual metrics, which are often difficult to translate into explicit descriptors of spatial configurations. Even when such metrics are available, their practical use as guiding tools in the design process is frequently obstructed by complex data interoperability procedures. These procedures are necessary to enable seamless data transfer across the multiple software environments involved in the design process. This study advocates for the softBIM paradigm, which optimizes workflows by embedding visual analysis results into target geometries. Supported by this process, the calculation of a metric to measure the impact of existing and planned visual obstructions on the vision of the targeted landmarks is proposed and analysed. This metric is specifically applied to assess the visual information incoming to the vertical facades of building envelopes, a context of application that denotes criteria of assessment different from the ones usually applied in the most established frameworks for visual analysis (e.g., isovist analysis). SoftBIM enables effective automation strategies to aid the metric computation and the processing of the results to implement seamless export and data implementation. The visual metric is built upon implementing the Ladybug suite and addresses the different limitations in the target-based visibility calculation supported by the tool.

Keywords: outdoor view; behavioural design; visual comfort; sustainable environment; field of view

check for updates

Citation: Cavaglià, M.; Speroni, A.; Blanco Cadena, J.D.; Mainini, A.G.; Poli, T. Exploring Built Environment Visual Interactions: A SoftBIM Data-Driven Approach for a Database About the Outdoor View. *Buildings* **2024**, *14*, 3340. https://doi.org/10.3390/buildings14113340

Academic Editor: Derek Clements-Croome

Received: 17 July 2024
Revised: 4 September 2024
Accepted: 26 September 2024
Published: 22 October 2024

1. Introduction

In recent times, the study of the visual characteristics of the built environment has gained increasing significance in the management of habitable spaces across different scales. Research in this field is extensive and remarkably diverse, encompassing a wide range of environments, regulations, and domains. Despite the variety, these diverse branches converge to establish a shared certainty: the significance of visual features within a space and the many benefits that proper consideration of these issues can bring to the human experience of the built environment.

In this regard, what is introduced here as visual characteristics can be broadly defined as everything that constitutes the flux of information that the people inside a given environment can acquire via the sense of vision. In particular, although visual information can be considered neutral, it may take on various meanings depending on the context. For example, visual information serves as a crucial conduit of data within the built environment for navigation and interaction. At the same time, different visual stimuli may also trigger diverse psychological and physiological responses, influencing the well-being of people via multiple levels of interaction. This complexity ultimately translates into a diverse array of specialized forms of visual analysis for the built environment, where depending on the unique type of visual interaction that needs to be addressed in a given situation, a different analysis framework must be implemented and deployed.

In this context, the study investigates a workflow to analyse the visual information incoming towards the vertical surfaces of buildings' facades. This choice is directly linked to the evaluation of visibility from the indoor spaces, as the visual information that can be captured from inside a building is filtered and mediated by its envelope. Therefore, the study of the visual flux directed towards the building envelope directly translates to the evaluation of the indoor visual access to the outside environment, a property of the habitable space often defined as the outdoor view [1].

The study's objective is to contribute to the advancement of quantitative visual analysis in the built environment by addressing key limitations found in existing visual analysis tools, such as the impact of the field of view constraints and the complexity of accessing visual data, which often impede the effective use and application of such data.

In this regard, it is developed an algorithm to compute a custom metric aimed at evaluating the reduction in potential visual access to specific landmarks within the site, both due to the effect of real or planned physical obstructions (i.e., other buildings and greenery) and the visual limitation acting on the field of view (i.e., limiting the visual cone depending to the direction of view). The study is developed using the widely-used Grasshopper Visual Programming Language (VPL) within the Rhinoceros v7 software environment. Additionally, this process is supported by the creation of a 3D visual database based on an entity-relationship (ER) model, which stores the results of the assessment within a relational database framework, enabling improved management and use of the data. This contribution identifies a workflow that leverages the synergy of different applications to enable a BIM-oriented strategy for data management, where the actual geometry in the CAD environment embeds the analysis results. A BIM-oriented methodology implies that all data directly linked to a specific environment, or technological component can be associated with the geometry representing that entity within the same file where site analysis or the design process occurs. This approach reduces the need to manage and integrate external resources, such as raster maps or external databases, which is typically critical due to their potential risk of generating errors (e.g., faulty data updates, complexity in exchange projects, etc.).

2. Visibility Assessment Procedures and Visual Data Output Storage

Visual studies have always been a fundamental area of interest when dealing with the manipulation of the built environment. While traditionally, these concepts were primarily explored in the context of the need to appropriately design the spatial configuration of habitable areas, encompassing functional and aesthetic considerations through qualitatively focused approaches, a new viewpoint has gradually emerged. This perspective aims to extract from the visual interaction different subsets of visual attributes, which can be quantitatively described to provide discrete backing for various analytical methods.

The most fundamental visual feature that can be quantitatively described is visibility. Visibility can be defined as the reciprocal quality of being able to see or to be seen from a specific location. Kevin Lynch was among the first to detail a few different concepts to implement visibility mapping of the built environment with specific goals. In this regard, it can be noted the concept of "visual absorption" (VA) is used to determine the degree to which an area can absorb transformations to its layout (e.g., new constructions or renovations) without apparent visible alterations (this is made possible by multiple geometrical factors like the irregularity of the topography, or the presence of visual obstructions like dense vegetation or urban canyons) [2] (p. 99). Lynch also presented the concept of "visual intrusion" (VI), which is the measurement of the visual field occupied by a target entity [2] (pp. 100–106). While these data can be utilized to compile multiple values at different viewpoints and visualize them across a designated area as value fields, a significant challenge in advancing visual analysis methods has been determining the appropriate methodological framework for collecting and measuring the required data. In this regard, Bittermann et al. mentioned that the geometrical characteristics of an environment can influence visual perceptions at a fundamental level [3].

In alignment with these observations, numerous frameworks for visual analysis primarily focus on two aspects: firstly, the computation of visibility, and secondly, the utilization of predominantly geometry-based approaches to accomplish this [4]. Existing procedures may vary depending on the scale of application (e.g., visual studies may be implemented across large territories, cities, lone districts, buildings, or even interior spaces) or the aim of the study (e.g., heritage protection [5], privacy control [6], comfort and well-being [7], or visual assessment [8]). Historically, the domain of territorial management has been one of the first actors to rapidly acquire and subsequently refine novel procedures for visual analyses of the built environment. In the context of environmental impact assessment (EIA), visual impact assessment (VIA) is routinely implemented to evaluate and identify the potential impact of transformation intervention in a given environment [9], following the trajectory set by Kevin Lynch and the concept of “visual absorption”. VIA produces reports known as Zone of Theoretical Visibility (ZTV) maps or Zone of Visual Influence (ZVI) [10] maps, which highlight the area from which an object may be theoretically seen [11]. These maps are mostly developed via Geographic Information System (GIS) applications from the study of precise elevation data acquired from digital terrain models (DTMs) or digital elevation models (DEMs) via a visual analysis process known as viewshed [9]. Viewshed analysis is a type of visual analysis used to compute the visibility across large territorial areas, and it is fundamentally related to isovist analyses [12,13]. Isovist and viewshed analyses were both derived from research developed in 1967 by Tandy [14]. A viewshed identifies the collection of visible locations from a specific vantage point. In this context, “visible” refers to unobstructed locations where a continuous line of sight can be extended to the point of view within a predefined distance threshold. GIS applications for viewshed analyses can also account for the earth’s curvature and atmospheric refraction, which is the bending of light rays due to variations in atmospheric density with height [11]. This contrasts with isovist analyses, which, due to the fact of being utilized at smaller scales (e.g., city, district, or interior spaces), do not necessitate these types of visual corrections. Instead, isovist analyses rely primarily on the same projective operations of perspective studies as described within projective geometry. However, viewshed analysis output is also dependent on the input data used to compute the results. As observed by Florio, due to the fact that the viewshed is built upon DTMs (Digital Terrain Models), DSMs (Digital Surface Models), or DEMs (Digital Elevation Models), elevation data, which are raster maps where each pixel stores an elevation measure of the territory corresponding to its position, the accuracy of the results depends on raster resolution, data sampling density, and interpolation algorithms [15]. In addition, the 3D model of the environment built upon DTMs, DSMs, and DEMs behaves as 2.5D models, being generated by vertically displacing a locally flat area, extracted from the earth’s surface, along the normal direction corresponding to each recorded height measure per pixel of the raster maps, essentially making impossible to further account for important details of the built environment along the vertical development of multiple structures, in particular the buildings (Figure 1) [16] (p. 3).

As previously mentioned, a more suitable visual analysis framework for the smaller scale of the built environment is the isovist analysis. Tandy authored this concept together with the viewshed, and while being similar, these analysis frameworks are usually regarded as separate entities due to their distinct usage practices [4] (p. 76). In particular, isovist analysis differs from viewshed analysis as it is usually performed upon vector reconstruction of space (both 2D and 3D). Isovist analysis starts by extending visual rays from an observation point until they intersect with boundaries. Professor Michael J. Ostwald & Professor Michael J. Dawes divide the analysis boundaries into four types: global boundary, visibility boundary, fixed boundary, and transient boundary [17]. While the global and visibility boundaries act as user-defined limits to enclose the region of space to be analysed, the visibility and transient boundaries represent instead physical entities located within said space that can block visibility. The only difference between the two is that visibility boundaries are static (e.g., walls, fences, trees, etc. . .) while transient boundaries are dynamic (e.g., doors, mobile shades, cars, objects in motion, etc.. . .). Each intersection point obtained via the "clash detection" between the visual rays and the boundaries is finally connected to form a so-called isovist polygon, which represents the spatial extension where unobstructed vision can happen. As with any other geometrical figure, it is possible to implement the isovist polygon to extract various dimensions in the form of lengths and areas, and the combination of this basic information represents the foundation for the definition of more complex indexes [18].

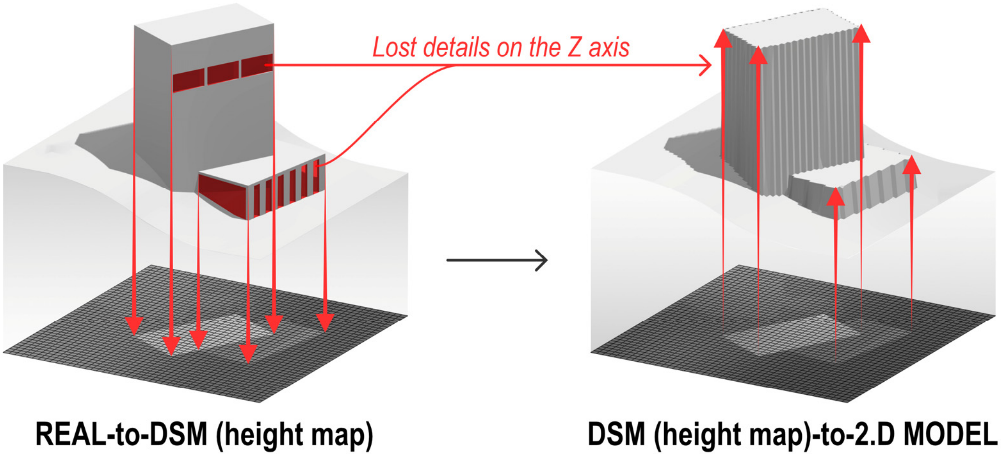


Figure 1. The figure further elucidates the 2.5D nature of DSMs in managing the geometrical information of real sites. On the right, the real objects and terrain constituting an environment are surveyed to store the elevation of each cell on a grid. This process generates a height map, represented as a matrix of height values across the XY plane. On the left side, the height map is accessed, and the stored height values are utilized to vertically displace a plane, thereby recreating a 3D model of the content. However, this transformation is unable to entirely replicate the real environmental model. Any geometric variations along the Z axis are effectively lost during the creation of the height map and thus cannot be accurately reproduced during the conversion to a 3D model.

Presently, the expected outputs of visual analyses primarily fit into one of the following categories: raster-based data mapping (e.g., heatmaps [19,20], data fields [13]), structured data formats (e.g., *.csv files [20]), or 2D/3D geometries (e.g., line of sights [19], viewshed surfaces [19,20], isovist polygons [21,22], Minkowski models [23]). Figure 2 visually details these concepts in a compact manner. This observation is consistent with the data export capabilities and options of the most implemented software environments dealing with visual analyses.

In this regard, Table 1 displays the digital applications reviewed within the current research framework. While the list includes some unique entries with notably different characteristics (e.g., the Visual Tracer addon [24]), most others propose similar approaches categorized as isovist-based, viewshed-based, or model-based, for instances that could not effectively fit into the previous categories and fundamentally relied on computing geometrical visibility checks using alternatives approaches.

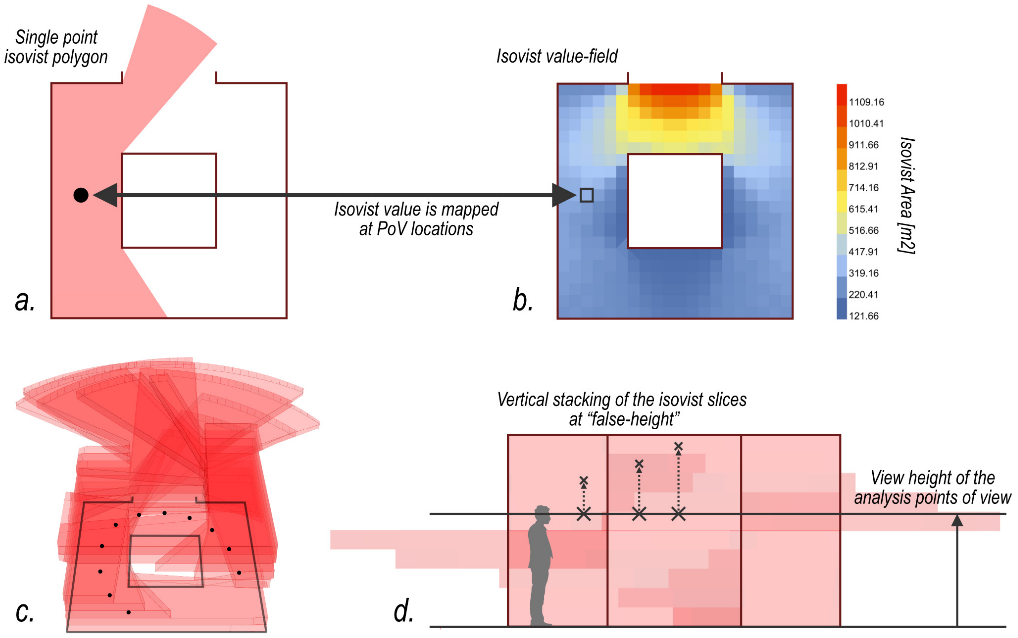

Single point isovist polygon
Isovist value-field
Isovist value is mapped at PoV locations
a.
b.
1109.16
1010.41
911.66
812.91
714.16
615.41
516.66
417.91
319.16
220.41
121.66
Isovist Area [m2]
Vertical stacking of the isovist slices at "false-height"
View height of the analysis points of view
c.
d.

Figure 2. Partial summary of data storage solutions for isovist analysis results. The list starts with the simplest instance: a single isovist polygon that, once saved, can be re-analysed for its geometrical properties to evaluate various visual features (**a**). Following this, a heatmap generated by placing the values stored within a structured data file and mapped as a spatial value field can be used to store the specific results describing selected visual attributes (**b**). Finally, Minowski models are presented, which geometrically store the isovist polygons from multiple points of view (**c**). To improve the comprehension of the Minkowski model configuration, the last schema shows the vertical stacking of the isovist slices at false height (**d**).

Table 1. Summarized list of the most notable software or add-ons that specifically implement visual analysis frameworks to extract various data at different scales of analyses.

Software Platform	Methodology Enabled	Type of Analysis	Supported Indexes or Tests	File Output	Reference	Last Update *	Release Date
Visual tracer v0.1 (Addon for Grasshopper)	Model-based Analysis **	3D	Eye tracking ** Preference maps **	3D point grid with associated data about line-of-sight intersection	[24]	2023	2023
Hawk v0.1 (Addon for Grasshopper)	Isovist-based analysis	3D	3D isovist volume	3D solid of an agent field of view (FoV)	[22]	2021	2021
Climate Studio v1.9	Model-based Analysis **	2D/3D	Quality views (LEED V4/V4.1, View credit)	List of values Vector and raster images	[25]	2022	2020
DeCodingSpace Toolbox v2021.10 (Addon for Grasshopper)	Isovist-based analysis	2D/3D	All the core indexes in the isovist analysis	List of values	[21]	2021	2019
Isovist v2.4	Isovist-based analysis	2D	All the core indexes in the isovist analysis	List of values Vector and raster images	[26]	2022	2017

Table 1. *Cont.*

Software Platform	Methodology Enabled	Type of Analysis	Supported Indexes or Tests	File Output	Reference	Last Update *	Release Date
Ladybug suite v1.8 (Addon for Grasshopper)	Model-based Analysis **	3D	Basic visibility checks * and LEED v3 standard output (View Credit)	List of values	[27]	2023	2013
DepthmapX	Isovist-based analysis	2D	All the core indexes in the isovist analysis	List of values Vector and raster images	[18]	2020	1998
ArcGIS Pro (ArcGIS 3D Analyst)	Viewshed-based analysis	2D/2.5D/3D	All the core indexes in the viewshed analysis	List of values Vector and raster images	[19]	2023	2015
QGIS v3.32.2 (Visibility Analysis plugin)	Viewshed-based analysis	2D/2.5D	All the core indexes in the viewshed analysis	List of values Vector and raster images	[20]	2023	2023

* Dates were retrieved on the same day as the reference cited in the respective column. ** Elements of the table were defined by the author to fill gaps in the reference information or to clusterize certain methodologies better.

In any case, despite individual differences, most of the reviewed environments offered a similar workflow to deal with the results of the visual analyses performed. In particular, the studied workflows treat visual analysis results as individual outputs with low integration between multiple outputs. For example, a viewshed analysis performed in GIS environments usually results in raster outputs where each cell records the results of said analyses. In this way, any change to the input parameters (e.g., the viewpoint position) generates a corresponding and independent raster file, separate from the previous one and archived onto a different dataset. At times, this makes it challenging to generate comprehensive visual analyses based on querying the existing results, particularly when multiple file outputs need to be integrated to access and compare various results simultaneously. As a result, sometimes, it is more efficient to set up and develop a different visual analysis from scratch rather than implementing the existing material into a custom workflow for querying its contents.

In this regard, the Rhinoceros platform, which presents multiple solutions for visual analyses via its vast array of addons for the plugin Grasshopper, suffers an even stricter regime for the integration of post-processing visual analysis results. This is because many add-ons, composed by scripting libraries with limited baking options, cannot natively bake certain types of data into the standard Rhinoceros environment. Such is the case of the Ladybug suite, where the results of the visual analysis components can be baked in the form of a coloured material library assigned to the tested geometries. Although the data is not effectively lost, it is nevertheless hardly implementable for subsequent evaluation besides visualization. Finally, it is worth noting that BIM-oriented software is not included in the list of tools reviewed in Table 1. Although BIM frameworks, such as Autodesk Revit [28] and Graphisoft Archicad [29], provide substantial value in managing and streamlining quantitative assessments—like those analysed in this study—they currently display minimal features or built-in capabilities for visibility analysis. Instead, these platforms typically depend on integrations with third-party applications or external plugins to perform such specialized tasks. Many reviewed platforms report compatibility with different BIM environments, particularly Climate Studio [30] and ArcGIS [31]. Therefore, despite BIM applications' highly peculiar working framework, visual analysis capabilities tend to overlap with the previously analysed perspective. That said, however, this study has noted how the exact workflow enabled by BIM applications may improve the implementation of visual analysis into broader operative workflows.

To conclude, the review of software enabling visual analysis revealed that many calculation frameworks limitedly account for the unique characteristics of indoor environments. Unlike outdoor analysis, where the observer can, and often needs to, actively scan the visual information all around, the indoor visual focus is more constrained, often limited to specific directions, such as the ones capturing windows (Figure 3). Consequently, factors like the human field of view play a more significant role indoors, yet many tools still need to address these variables fully. This study aims to integrate these considerations to develop more realistic assessments.

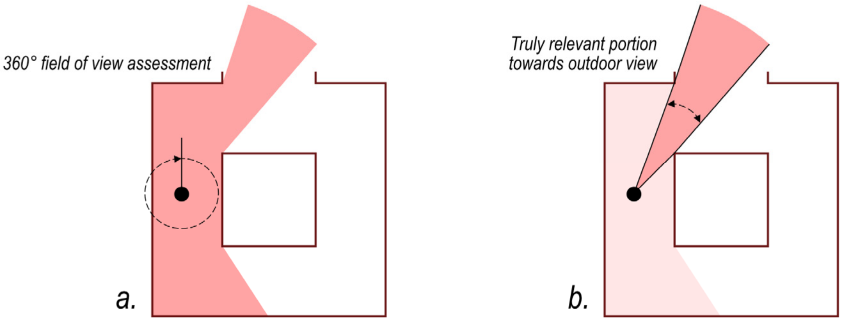


Figure 3. The image summarises the characteristics and the relevance of the visual data acquired by the isovist analysis initially presented in Figure 2. While isovist analysis can output many metrics based on the visual data acquired from around the observer location (**a**), the information more relevant to describing outdoor view is limited to a smaller field of view (**b**).

3. Human Vision and the Characteristics of the Visual Field

The phenomenon of human vision is a result of light rays being reflected in the environment and entering the eyes. Incidentally, among all the possible light rays travelling in a given environment, only the subgroup able to reach, via an unobstructed path, the eyes can be elaborated to develop the phenomena of vision. Therefore, only the information carried along an unhindered trajectory can be treated as visible to the visual experience. The array of visible light rays identifies a spatial volume defined as "visual field". The visual field (VF) is defined as "what can be seen when head and eyes are kept fixed [32], and although fundamentally different, it can be likened to the concept of "field of view" (FOV) applied to the study of optical sensors and instruments (e.g., photography), which is a solid angle through which a detector is sensitive to electromagnetic radiation [33]. As a result, the visual field can be measured as the angular span from the observer's line of sight (LOS), within which stimuli from the external environment may develop perceptual experiences [4].

However, the analogy between the human visual field and a camera's field of view becomes less accurate as the geometrical properties of the two are further detailed. For instance, a standard camera can typically achieve a variable focal length using different zooming devices, whether they are digital or optical. Since the visible range of the field of view is directly linked to the instrument's focal length, the dimensions of the field of view can dynamically vary. On the other hand, the overall structure of an individual's eye is relatively stable [34], which results in firmer limits for the human visual field.

The visual field of a typical human eye develops across a vertical span of 120 degrees and an almost 160-degree range horizontally. However, neither dimension is symmetrically distributed around the eye center. Starting from the fixation point (e.g., the visual target at the end of the line of sight of a singular eye), stimuli are usually detectable up to 60 degrees above and 70 degrees below, 60 degrees inward (nasally), and 100 degrees outward (temporally) [35]. The aforementioned visual field, being limited to a single eye, is usually addressed as a "monocular visual field" or "monocular field of view". However, the visual field deployed by each eye of the pair overlaps with one another in the region centred around the line of sight, defining the binocular visual field. The binocular field of vision extends for a horizontal range of around 120 degrees centred around the line of sight and represents the spatial extent where binocular vision is enabled [36] (p. 12). Binocular vision also referred to as "eye teaming" [37], defines the process through which both eyes collaborate to combine the images perceived by each eye into a single integrated image.

Furthermore, it is important to clarify how the visual capabilities of human vision are not uniform within one's visual field. The visual sensitivity of the human eye is directly linked to its physiology, particularly the internal distribution of the photoreceptor cells (equivalent to the camera sensor) [38] (pp. 274–280). In this regard, Grandjean Etienne's work about ergonomics in the workplace detailed three concentrical regions of the visual field determined by the angular distance from the line of sight. Among these, the optimal visual field ranges from 0 degrees to 1 degree, the middle visual field ranges between 1 degree and 40 degrees, and the outer visual field ranges between 40 degrees and 70 degrees [32] (p. 234). Specifically, clear vision happens in the optimal visual field and partly in the middle visual field, whereas the outer visual field can retain just the necessary visual sensitivity to notice objects in motion. While Grandjean proposal may appear inconsistent with other findings [39], it is noted how the ergonomic-related application of the proposed work is well-aligned with the study of the built environment and its characteristics; for this reason, it is proposed that the generalisation introduced are suitable for the application of this model in the present context.

While not all the presented physiological constraints have been historically addressed within the many frameworks for visual analysis in the built environment, the different features outlined in the visual field are nevertheless recognized as a necessary background for the development of conscious and proper visual analysis procedures. It is also worth mentioning that while all the provided information for understanding and accurately representing a person's visual field has been confined to a static state, more intricate nuances can emerge when considering movement (including both eye and body movements) to develop sequences of more intricate postures (such as sitting, standing, working, etc.).

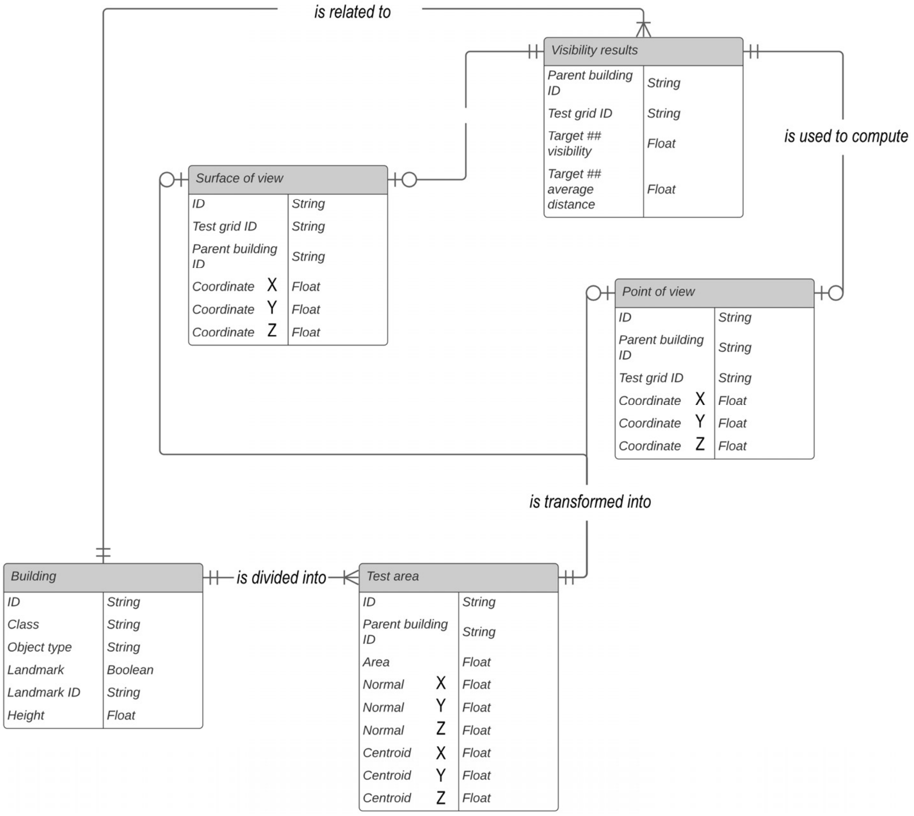
4. Methodology

The addon Elefront [40] was utilized within a Grasshopper script to implement the paradigm of softBIM to achieve the study objectives outlined in the introduction. This approach embedded the results of visual analyses directly into the data structure of 3D models in Rhinoceros by storing outputs as simple key-value dictionaries. This approach enabled the effective recording and organization of visual analysis data, ultimately creating a visual database for storing the acquired information.

Fagerström et al. proposed in 2012 the concept of "*softBIM*". SoftBIM is the implementation of BIM processes (e.g., the embedding of metadata within geometric models) within non-BIM software environments via the use of custom-made code interfaces [41]. Elefront can be regarded as a softBIM enabler technology that uses the attribute user text data space to store custom-coded data from Grasshopper scripts [42] into the geometries baked into the Rhinoceros scene.

The softBIM paradigm is applied in this study to set up a work environment capable of receiving and processing visual analysis with increased efficacy. This is relevant because one significant challenge in traditional CAD environments is the difficulty of integrating and managing value lists that contain data entries related to the geometries within the 3D scene. Exporting this data while ensuring it remains correctly associated with the relevant geometries often requires complex workflows and significant manual intervention with the potential risk of introducing errors. SoftBIM applications are key to integrating analysis results and design data into a cohesive environment. The coupling of geometries and analysis data also established ideal conditions to favor the automation of the exchange of information across different stages of the design workflow. As a result, SoftBIM significantly reduces the complexity and potential errors associated with managing and transferring external data sources, streamlining the workflow, and enhancing collaboration between different software environments and potentially different actors.

Based on the possibilities of the softBIM paradigm, a simple database structure has been conceived to organize the Rhinoceros file data structure to store the outcomes of visual analyses performed via Grasshopper scripts. The database schema displayed in Figure 4 shows the different entities selected to be part of this definition. In this case, each entity refers to geometry-type entities containing a series of key-value data pairs into the corresponding Attribute User Text data space within the Rhinoceros scene.



Visibility results	
Parent building ID	String
Test grid ID	String
Target ## visibility	Float
Target ## average distance	Float

Surface of view	
ID	String
Test grid ID	String
Parent building ID	String
Coordinate X	Float
Coordinate Y	Float
Coordinate Z	Float

Point of view	
ID	String
Parent building ID	String
Test grid ID	String
Coordinate X	Float
Coordinate Y	Float
Coordinate Z	Float

Building	
ID	String
Class	String
Object type	String
Landmark	Boolean
Landmark ID	String
Height	Float

Test area	
ID	String
Parent building ID	String
Area	Float
Normal X	Float
Normal Y	Float
Normal Z	Float
Centroid X	Float
Centroid Y	Float
Centroid Z	Float

Figure 4. Database schema proposed to manage visual analysis within the Rhinoceros v7 software environment.

Figure 5 displays the representation of the geometry instances related to the proposed schema using an application case developed at the city scale with the focus of mapping the visual information incoming onto the facade surfaces of buildings. While the proposed schema could be applied to visual analysis at various scales—from the interiors of buildings to the expansive visual coverage of large territories—this study focused on a district-scale test run. This scale of study was chosen because it is considered an optimal setting to assess the system's ability to accurately represent the visual conditions of potential building sites. What is presented here as the entity "visibility results" is intended to represent a reference geometry containing a value list of the results of visual tests performed via a visual simulation built upon a specific viewpoint. Said viewpoint can either be determined by the geometrical data of the "point of view" or "surface of view" geometries, which in turn represent the analysis grid built upon the surfaces to be analysed. Figure 6 summarises in a more detailed manner the subdivision of facade surfaces into analysis cells. Each analysis cell constitutes the anchor point for a point of view.

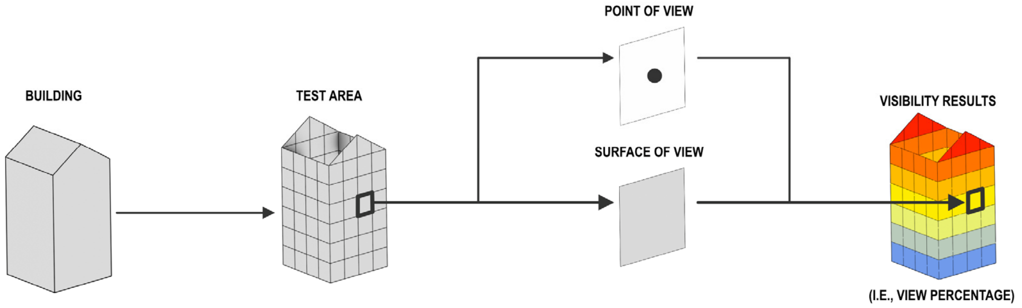


Figure 5. Highlight of the geometry linked to each entity of the proposed database structure.

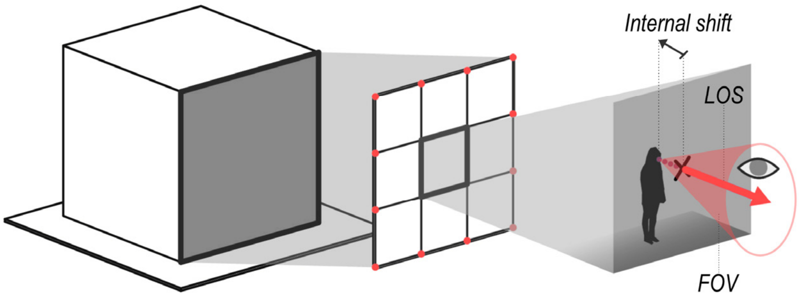


Figure 6. Schematic process for the development of the basic visual constraints considered within the proposed visual analysis workflow: from left to right, the basic geometry of the building's volume is subdivided into a sample grid composed of cells with equal sides (as much as possible). The central point of each cell is then used to place a point of view for testing. Each point of view is further associated with an internal shift, a line of sight (LOS) with a specific direction, and a field of view (FOV) with a custom angle. The data about LOS and FOV is uniformly applied to all points of view within the analysis.

The construction of the point of view, which is based on referenced data, can be further refined by integrating additional data about potential visual constraints. This enhancement increases the value of the visual simulation. Among the basic data, an "interior displacement" can be established to shift the point of view to the inside of the building's geometry. This shift simulates a more realistic visual condition by analysing the visual information incoming to an internal location within the building, as opposed to using a generic and less realistic position derived from the plane of the facade. Simultaneously, a line of sight (LOS) direction and a view angle can be implemented to filter visible objects along a specific direction within a given field of view (FOV). In this context, each point of view is studied using a line of sight normal to the analysis cell's central point and a viewing angle equal to 60° to frame a visual area with higher view quality. This information is also stored within the point of view and surface of view entity.

In this regard, it is important to define which kind of visual data will be computed within the visual analysis. In the domain of visual analysis, it is possible to outline two main methodological approaches for data extraction. In this context, it is proposed that these approaches be named with the generalized categories of visual field analysis and targeted entity visual analysis. The former (e.g., viewshed analysis and isovist analysis) focuses on the overall content of visual information perceived within a certain perspective, while the latter (e.g., visibility analysis) focuses on the visual perception of specific objects within that field. Figure 7 further elaborates on the particulars, showcasing how these approaches may output different types of data from the same visual environment.

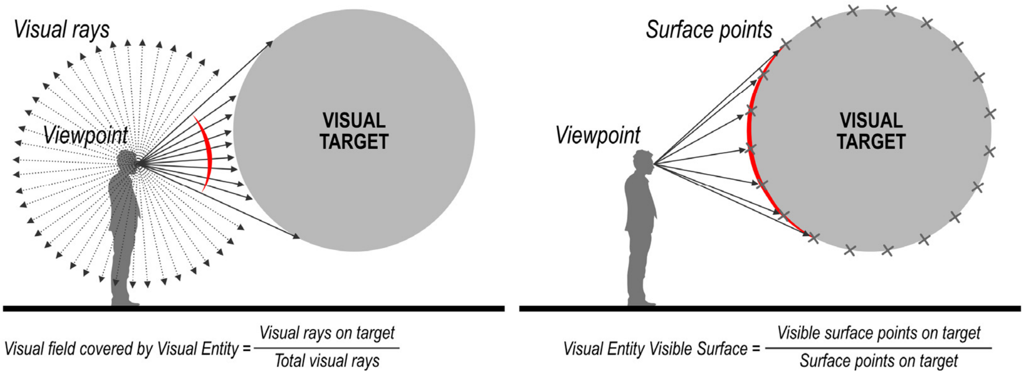

Visual rays
Viewpoint
VISUAL TARGET
Visual field covered by Visual Entity = Visual rays on target / Total visual rays
Surface points
Viewpoint
VISUAL TARGET
Visual Entity Visible Surface = Visible surface points on target / Surface points on target

Figure 7. Examples of different approaches that can be implemented in visual analysis: On the left, visual field analysis focuses on the information present within one's field of view. On the right, targeted entity visual analysis concentrates on the visual conditions under which a given entity is perceived (for example, how much of the entity's surface is visible or not).

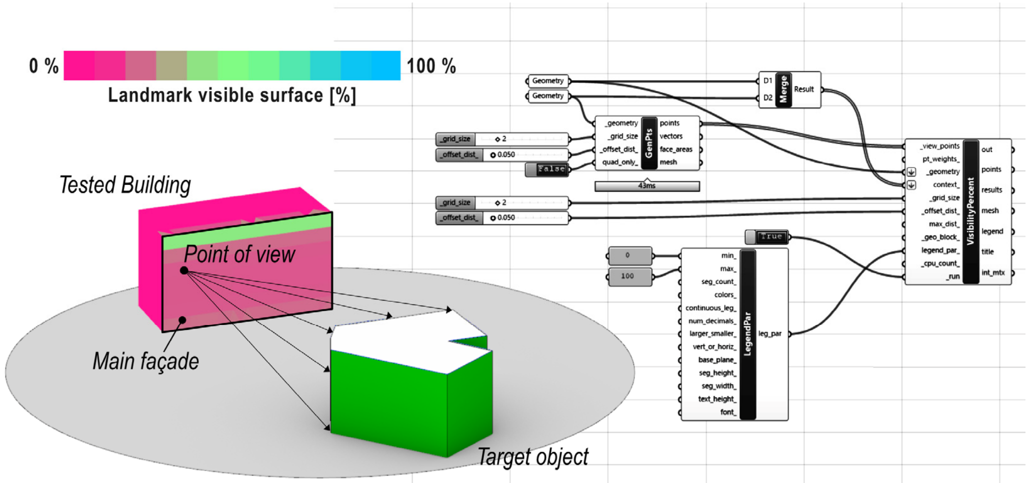
The visual metric, which is analysed in this contribution, evaluates the reduction in potential visual access to specific landmarks within the site as viewed from a building envelope surface. It is, therefore, a type of targeted entity visual analysis. It is noted that within the framework of Grasshopper, a similar assessment can be performed by the visibility percent function available in the Ladybug Suite [43,44]. The visibility percent tool, while helpful, has several limitations that hinder its ability to capture complex visual conditions fully. This tool calculates the percentage of an object's surface visible from a specific point of view, and it is linked to the observed entity rather than the observer. The output is a percentage that indicates the amount of visible surface area over the total outer shell surface of the object. For instance, a 50% value suggests that half of the object's surface is visible from the given viewpoint. However, achieving 100% visibility is impossible for closed solids because the front and back of the object cannot be viewed simultaneously. This leads to a case-by-case peak-value variation, making comparative analysis challenging. Furthermore, the tool does not characterize the analysis with a specific field of view or line of sight; instead, it captures data from all directions, complicating its use in scenarios involving multiple objects from an indoor perspective. Consequently, the tool's application can provide insight into the site's visual characteristics but is not sufficient to develop a comprehensive evaluation. Figure 8 displays a visual analysis developed to test the visibility percentage of the target object from a nearby building. The scene does not provide any kind of visual obstructions between the two volumes. In addition, as the data was produced by testing all possible lines of sight, no matter how sloped from the building façade, it is also possible that while a point recorded high visibility, such visibility was a consequence of accessing the view toward the target from an unrealistic viewing angle.

The metric proposed here is built upon the visibility percent computing to support it with more extensive visual analysis within the Rhinoceros environment. The metric is designed to meet three characteristics:

- The capability of recording a homogenous peak value to describe optimal visual conditions of target objects without obstructions;
- The capability of accounting for a limited field of view set around a main LoS direction;
- The capability of mapping results onto a homogeneous domain to simplify the comparison of different analysis results.

This metric is calculated within the script as the percentage of the visible surface over the potential maximum visible surface resulting from the absence of any obstruction (besides the ground geometry). This data can be defined as the accessibility of viewing a certain landmark (or any other kind of entity). In this regard, it would be possible to record a peak value of 100%, indicating that the entire potentially visible surface of the landmark is visible from a specific point of view. Conversely, a value of 0% means that the obstructions in question completely block the visibility of that potentially visible surface.

Essentially, any time the value of landmark visibility decreases from the ideal maximum value of 100%, it corresponds to the negative obstruction effect of visual obstacles placed along the LoS that connect the point of view to the landmark. These obstructions can be either existing or planned, so the metric can evaluate current site conditions or propose future predictions tied to site transformation. Figure 9 further emphasizes this setting. To properly weigh this data, the average visual distance of the object from the reference point of view is saved and stored as a complementary parameter. Figure 10 displays an instance of visual computation for a given point of view.



Maximum Visibility Percent	Minimum Visibility Percent	Average Visibility Percent (Whole Building)	Average Visibility Percent (Main Façade)
41.66%	0%	10.79%	22.02%

Figure 8. The image presents a script definition for performing the visibility percent analysis. The obtained values depend highly on the target object geometry, making comparing different conditions hard to implement. Also, the analysis does not filter visual directions that, while being viable, may not be framed within common visual conditions. For example, the highly sloped view directions linked to the point of view represented are all equal in the calculation, even though some may require very specific indoor layouts to be made accessible.

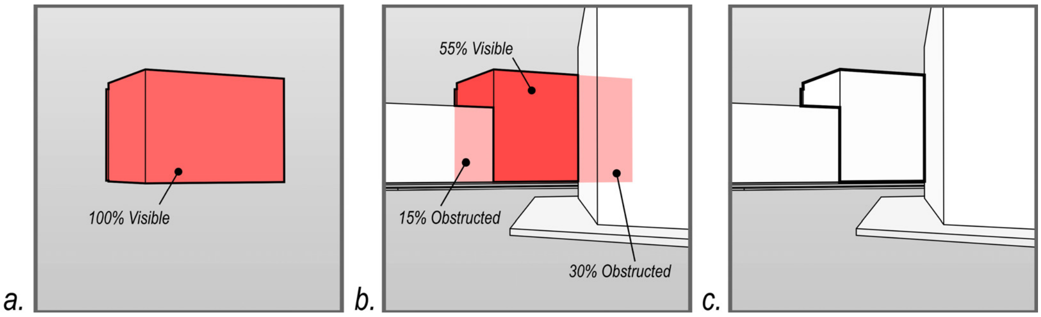


Figure 9. The figure further elaborates on possible data developed within the application of criteria for targeted entity visual analysis. In this case, considering an ideal, undisturbed visual condition without any kind of obstruction, the resulting visible surface of an entity can be treated as the maximum possible amount available for viewing (**a**). By adding the interference of physical obstructions or visual constraints such as fields of view (FOVs), the resulting visual access may diminish (**b**). This comparison between ideal and real conditions may serve to assess the percentage impact of visual obstructions or constraints in occluding visibility towards an entity (**c**).

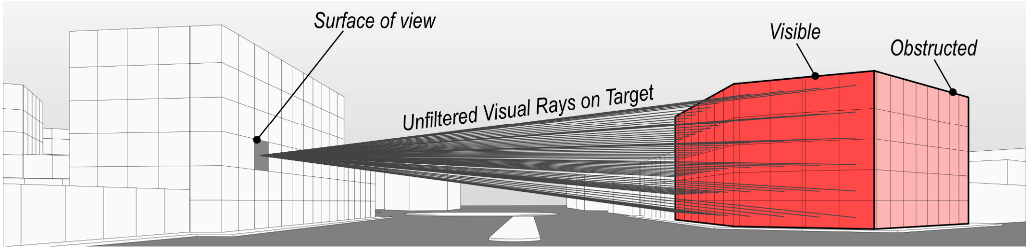


Figure 10. Clash detection is implemented in this instance to sort visual rays with clear access to the view. As visual rays represent both a vector and a segment, an average of their length can be computed to measure the average visual distance of an entity from the referenced point of view.

The complementary use of visibility and distance allows the proper weighing of instances where full visibility of distant objects or limited visibility of near objects may be recorded. By weighting these instances appropriately, it becomes possible to create a basic yet accurate assessment of the landscape.

In the given example, the list of landmarks is considered a subset of the building list. Therefore, if a building is also identified as a landmark for visual study, it will be assigned a unique landmark ID.

When conducting visual analysis using virtual models, it is important to recognize that the insights gained are inherently limited by the level of detail present in the digital models themselves. Llobera methodically presented a model to properly manage visual analyses in the form of the concept of the "Visualscape" [45]. A visualscape is defined as the spatial representation of any visual property generated by a spatial configuration. Essentially, to accurately assess the relevance of any visual property or data derived from analysis and presented in any form of representation, it is also necessary to consider a third element linked to this system: the "spatial configuration". The concept of visualscape is fundamentally centred on the spatial configuration. This means that by altering the selection of spatial components that constitute a spatial configuration or virtual model, it is possible to effectively change the scope, scale, and outcomes of the visual analysis.

Indeed, varying the level of detail in a virtual model can change the results obtainable from a visual analysis. For example, building models may be simplified to prismatic volumes, or they may include geometrically accurate sloped roofs, correctly displaced facade geometries, and so forth. Therefore, selecting a certain level of detail and coupling this data to a visual analysis report is an important procedure to facilitate the understanding of the visual data compiled. To this end, the present work implements the LODs system described within the OGC standard CityGML 3.0 [46].

In the following sections, the conceptual functioning of the algorithm used to calculate landmark visibility and accessibility is explained in greater detail. A test case is applied to demonstrate the algorithm's performance and capabilities. The test case features a simplified urban model designed with Level of Detail 1 (LOD 1) [47], representing various urban settings, such as narrow and wide streets lined with buildings and an open area akin to a plaza or a city's edge. Multiple buildings of different sizes and heights populate the model, creating a diverse urban environment.

Within this scene, two specific buildings have been chosen as landmarks for the visual analysis: one located in an open area with relatively unobstructed views and another situated deep within a narrow road. The algorithm will evaluate visual accessibility by analyzing the visibility of these landmarks from the envelopes of all other buildings in the scene. The visual analysis simulates realistic conditions by incorporating a field of view (FoV) of 60° (30° per side of the line of sight) and utilizing an internal offset of 15 cm from the building envelope surfaces to ensure accuracy in measuring visibility. This test case exemplifies the algorithm's ability to handle diverse urban conditions and assess how obstructions and location impact visual access to key landmarks.

Figure 11 provides a more in-depth breakdown of the script's execution and relative outputs. Each major entity of the database is categorized and stored in an individual dataset.

The sum of individual datasets composes the database structured in the Rhinoceros file. In essence, all datasets corresponds to a container for specific geometries and related embedded data. All datasets are generated from a linked individual process, which implements specific inputs. In particular, aside from the first dataset, which is built upon GIS cartographic data, all subsequent datasets are built upon the data stored in the previous dataset.

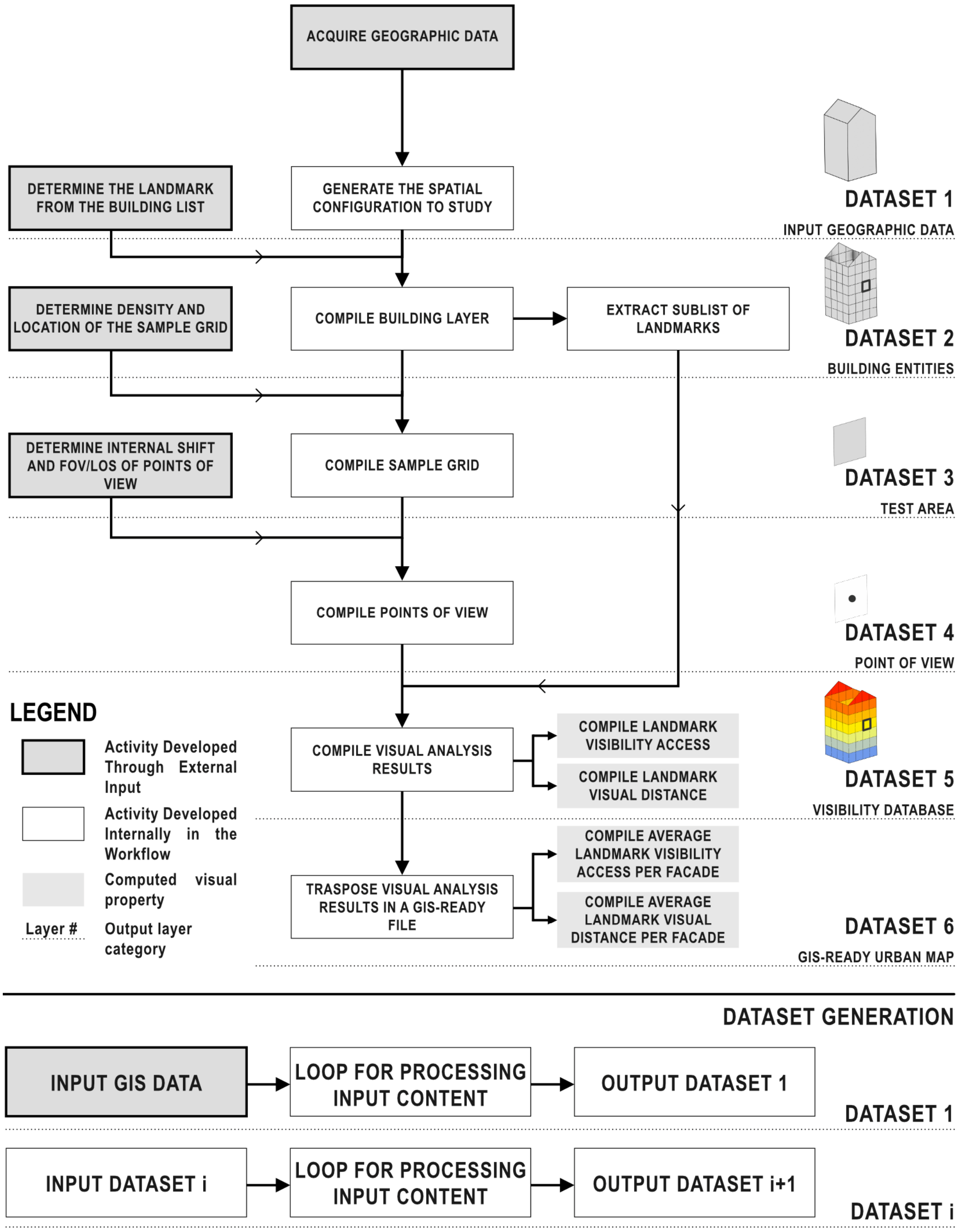

ACQUIRE GEOGRAPHIC DATA
DETERMINE THE LANDMARK FROM THE BUILDING LIST
GENERATE THE SPATIAL CONFIGURATION TO STUDY
DATASET 1
INPUT GEOGRAPHIC DATA
DETERMINE DENSITY AND LOCATION OF THE SAMPLE GRID
COMPILE BUILDING LAYER
EXTRACT SUBLIST OF LANDMARKS
DATASET 2
BUILDING ENTITIES
DETERMINE INTERNAL SHIFT AND FOV/LOS OF POINTS OF VIEW
COMPILE SAMPLE GRID
DATASET 3
TEST AREA
COMPILE POINTS OF VIEW
DATASET 4
POINT OF VIEW
LEGEND
Activity Developed Through External Input
Activity Developed Internally in the Workflow
Computed visual property
Layer #
Output layer category
COMPILE VISUAL ANALYSIS RESULTS
COMPILE LANDMARK VISIBILITY ACCESS
COMPILE LANDMARK VISUAL DISTANCE
DATASET 5
VISIBILITY DATABASE
TRASPOSE VISUAL ANALYSIS RESULTS IN A GIS-READY FILE
COMPILE AVERAGE LANDMARK VISIBILITY ACCESS PER FACADE
COMPILE AVERAGE LANDMARK VISUAL DISTANCE PER FACADE
DATASET 6
GIS-READY URBAN MAP
DATASET GENERATION
INPUT GIS DATA
LOOP FOR PROCESSING INPUT CONTENT
OUTPUT DATASET 1
DATASET 1
INPUT DATASET i
LOOP FOR PROCESSING INPUT CONTENT
OUTPUT DATASET i+1
DATASET i

Figure 11. The image represents procedural flowcharts describing the different processes composing the algorithm described in the study. The top part of the image specifically displays the workflow of the script execution. On the right, each "DATASET" label represents the corresponding list of geometries stored within a separate layer within the Rhinoceros file. The lower section of the image summarizes the two main computing processes used to generate the datasets. All datasets are produced through a looped calculation, applying a set of operations to the input data. Dataset 1 is created by processing external GIS cartography, while subsequent datasets are generated by iteratively processing the output of the previous dataset.

In this regard, Figure 12 further displays a general outline of the script complexity, while Figure 13 illustrates an example of the data linked to a geometry belonging to the category Dataset 5—Visibility database output, which stores the visual analysis results.

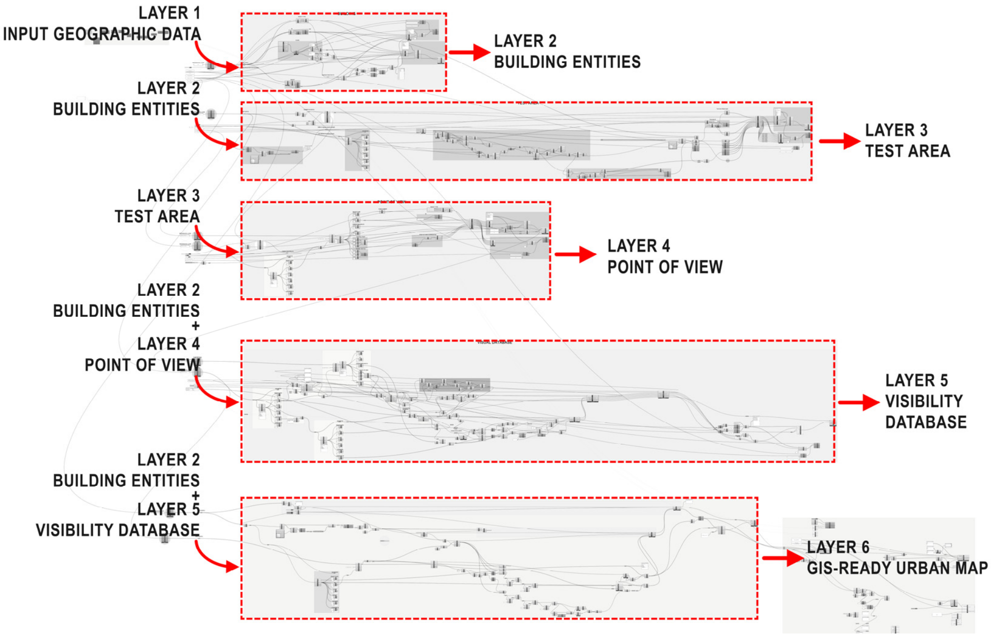
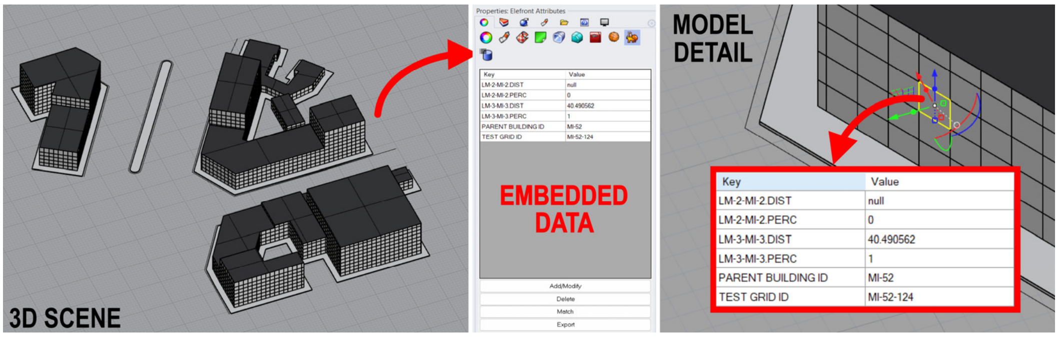


Figure 12. The image displays an overview of the script size and complexity. The algorithm, developed within the Grasshopper VPL, is segmented into distinct groups/definitions, each tasked with processing and generating a specific dataset. These datasets consist of geometrical objects containing various types of directly embedded data.



Key	Value
LM-2-MI-2.DIST	null
LM-2-MI-2.PERC	0
LM-3-MI-3.DIST	40.490562
LM-3-MI-3.PERC	1
PARENT BUILDING ID	MI-52
TEST GRID ID	MI-52-124

Figure 13. Example of the output from data dataset 5, the "visibility database". This dataset stores each grid cell utilized for visual analysis, with each geometry embedding a list containing various outputs from each visual query stacked within the script. In this particular case, the key tag for visual analysis results is constructed by concatenating the landmark code with the type of output (e.g., 'DIST' for average visual distance and 'PERC' for visibility percentage). All analysis results are cumulatively saved in the same file.

The list of values that stores the outputs of the visual analysis (i.e., Dataset 5—Visibility Database) is based on three main pieces of information: the geometry of the sample cell extracted from the analysis grid, and two string data elements, namely the test grid ID and the parent building ID. Upon this foundation, an arbitrary number of results can be added. The keys for each result are named by concatenating the landmark ID, which refers to the value, with a suffix identifying the type of data.

Developing processes with a high degree of automation has enabled the addition of an arbitrary number of results, allowing for the testing of as many landmarks as needed without requiring manual adjustments or accommodations in the script execution. In this study, automation is pursued by organizing data streams to collect the results of unique analyses, such as the visual percentage or visual distance of landmarks, into a single unified flow—regardless of the number of iterations. Each stream is implemented in the "Define Object Attributes" function from Elefront by matching the Keys and Values to implement as the embedded data list (Figure 14).

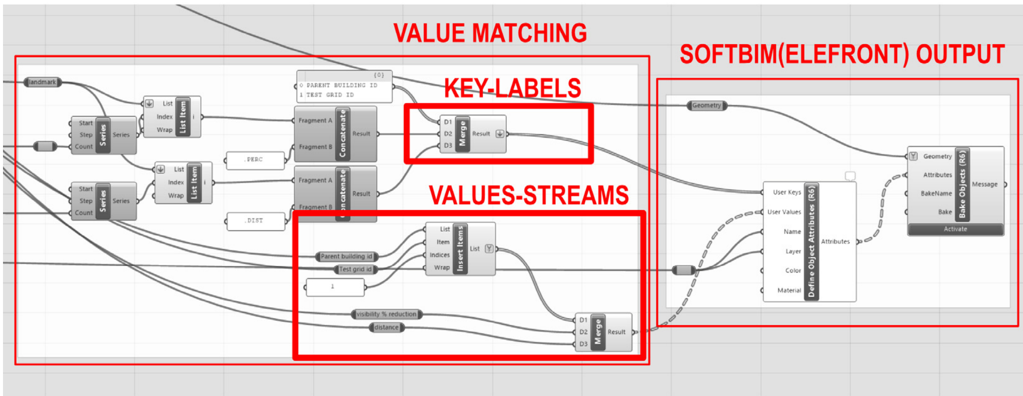


Figure 14. The figure details the endpoint of the script used to generate the content of Dataset 5—Visibility database. Each analysed metric being independently calculated based on each different landmark can output a variable number of results. In this instance, each unique metric is managed to develop a lone flux of data within the Grasshopper script. Since the connector transferring the results is only one, it is more efficient to implement this in further functions. The image also shows the functions of Elefront (i.e., "Define Object Attributes" and "Bake Objects"), which, when combined, allows the output of the softBIM-ready dataset.

The database is structured in a flexible manner, enabling efficient management of the recorded data. It allows for updates with results obtained from the analysis of additional landmarks or the computation of different information.

Figure 15 showcases the application of the said process to study the visual accessibility to the two separate landmarks determined within the test site developed in the study.

Despite the simplicity of the scene, the data mapped onto the building envelopes display the inherent complexity of the flux of visual information and its accessibility.

The implementation of the softBIM approach has enabled the integration of intricate and diverse data sets within a unified software environment. By leveraging the strengths of different platforms and domains, the softBIM approach streamlines the process of incorporating and managing complex visual information, ultimately enhancing the efficiency and effectiveness of building facade design and analysis.

The final phase of the current project focused on the design of a database for the softBIM prototype workflow; it addressed the crucial step of exporting the collected information into a GIS environment. This integration is vital as it facilitates the spatial analysis of data within a more comprehensive, geographically contextualized framework, enabling the potential for multi-scale data implementation in the management of the built environment. However, this process presents challenges, particularly given the 3D nature of the database constructed in Rhinoceros. Key issues include ensuring compatibility between the complex 3D data of Rhinoceros and the GIS platform, accurately maintaining the integrity of spatial data during the transfer, and efficiently processing the large volumes of data involved in 3D models in the usually much more compact and streamlined modelling managed within GIS environments. Addressing these challenges is essential for the seamless integration of BIM data into GIS, unlocking new dimensions in spatial analysis and design. The actual export of the data in a georeferenced shapefile format (SHP) has been handled within the Grasshopper script via the function "Export Vector" [48] from the Heron addon [49].

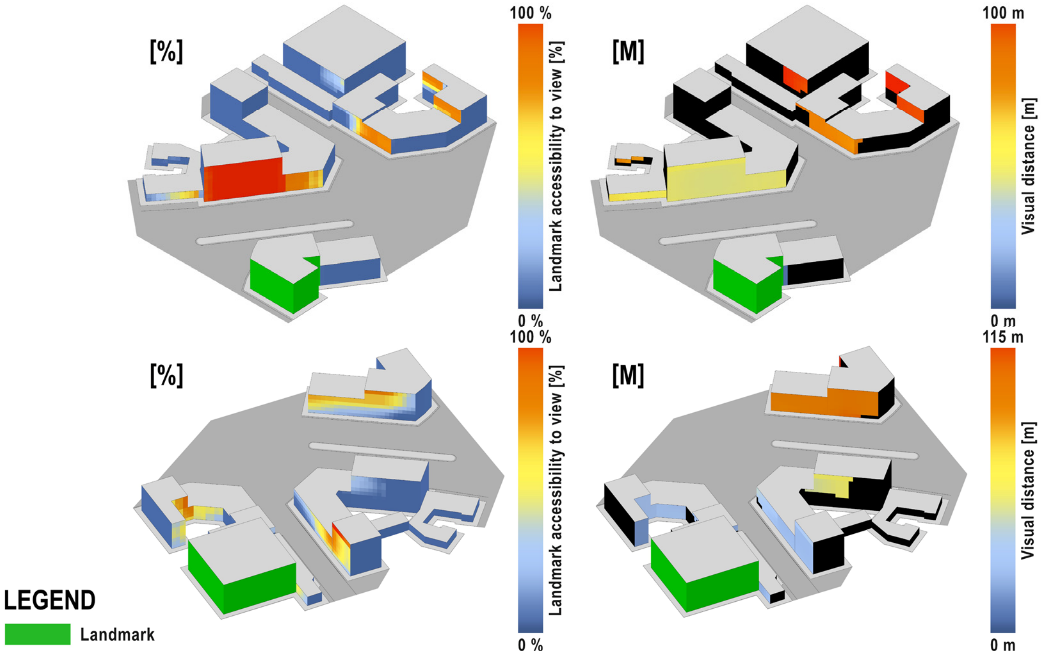


Figure 15. An example of data mapping illustrating landmark visibility includes the view accessibility ratio and the average visual distance to the landmark (green) from the vertical surfaces of nearby buildings. The visibility of a landmark is impacted by visual obstructions and the set field of view limit. This means that the assessed value may change even if there are no visual obstructions present, especially if the landmark is so close that it cannot be fully seen within the field of view limits. In that instance, the ratio gets lower due to recording the "virtual obstruction" of the visual clipping due to the field of view.

While the handling of 3D spatial data has been gaining increased support within GIS applications over the years [50], it is noteworthy to mention that transposing complex information into 2D geometries remains a fundamental step in generating clear and readable urban analyses. In this regard, a method is proposed to compress visual analysis results derived from facade analysis into the 2D shape of a building. This is achieved through a schematic representation based on the sectorization of the building's perimeter using its medial axis. The medial axis of a perimeter is the set of all points having more than one closest point onto said perimeter [51]. Starting from the building perimeter vertices, which can be organized by storing them in a unique list based on the type of perimeter discontinuity, the medial axis can be implemented as a spine to partition the overall building perimeter into multiple polygons, one for each facade edge.

The final GIS output exports the obtained polygons, assigning to them a single compact average of the results from all the sampled points on the facade for each analysed parameter. Visual-based indexes are averages of all viewpoints data, weighted by the sample grid cell area of each. Instead, distance-based information is considered the average of the sole sampled viewpoints with actual visual access. In this way, it is possible to assess a clear picture of the overall visual access of the facade to external contents while simultaneously understanding the average visual distance of these contents from the facade surface with available visual access. Figure 16 presents the main steps of the algorithm exporting execution.

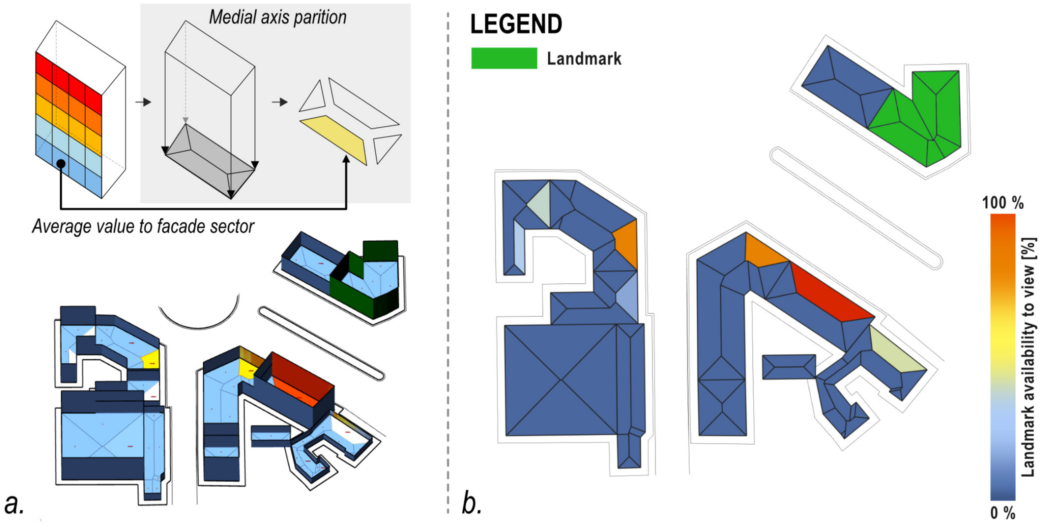

Medial axis parition
Average value to facade sector
a.
LEGEND
🟩 Landmark
100 %
Landmark availability to view [%]
0 %
b.

Figure 16. The image presents the process of extracting a 2D representation of a building element with partitioned geometry suitable for hosting average data of the entire façade. On the left, we observe the progression towards planar projection sectorization (**a**), representing a pivotal step in visualizing and interpreting the spatial distribution of the analysis outcomes. On the right, the output extracted from the Rhinoceros environment demonstrates seamless integration into the GIS environment (**b**).

In summary, it is worth stressing that careful implementation of FoV evaluation filters may help to overcome potentially misleading observations about visual relationships that may arise when such parameters are not adequately addressed. Figure 17 emphasizes this idea by illustrating how the tested façade, constrained by a narrower visual field, cannot establish a visual link with the analysed landmark.

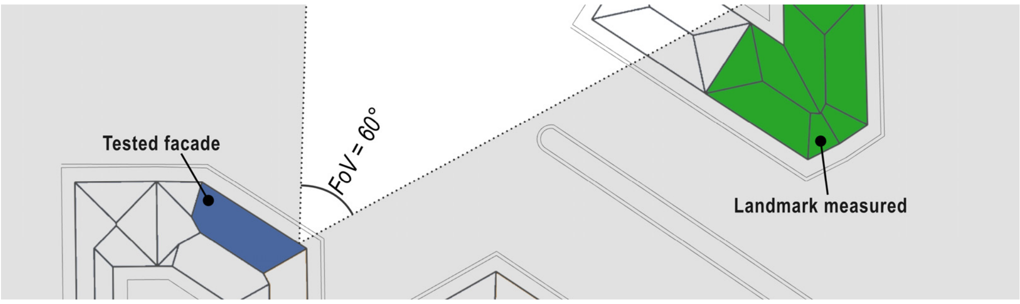


Figure 17. The figure details the influence of the constraints of the field of view on effectively computing the actual visibility of landmarks. In this instance, the tested facade intuitively appeared able to visually perceive the landmark. However, considering a more limited field of view, which may result from both biological considerations and structural interferences caused by the dimensions of windows hosted in the facade, the landmark may be invisible.

5. Results

The workflow developed can ultimately be implemented to create a softBIM file that stores various types of visual analysis results for further implementation or interoperability with subsequent stages in design endeavours.

The same process has supported the application's use of a landmark accessibility ratio. Segmenting the analysis process into individual automated steps enabled by softBIM data exchange among output datasets increases the computation complexity and length of the designed processes while ensuring faster data updates or implementing common steps across different evaluations. Figure 18 displays a potential application of an automated update process to display predicted impacts in landmark visual access for different types of transformation processes. In contrast to the visibility percentage analysis shown in

Figure 9, the custom metric provides a more intuitive representation of overall visual access to the landmarks. Under optimal conditions, the main façade achieves nearly 100% visibility across most of its surface, with only a slight reduction at the outer edges due to the angular limitations of the field of view (FoV). Additionally, reductions in visual access caused by transformations are more clearly reflected in the assessed values. For instance, transformation (b) in Figure 19 results in an average reduction of 47% in visual access, while transformation (c) records a 35% reduction.

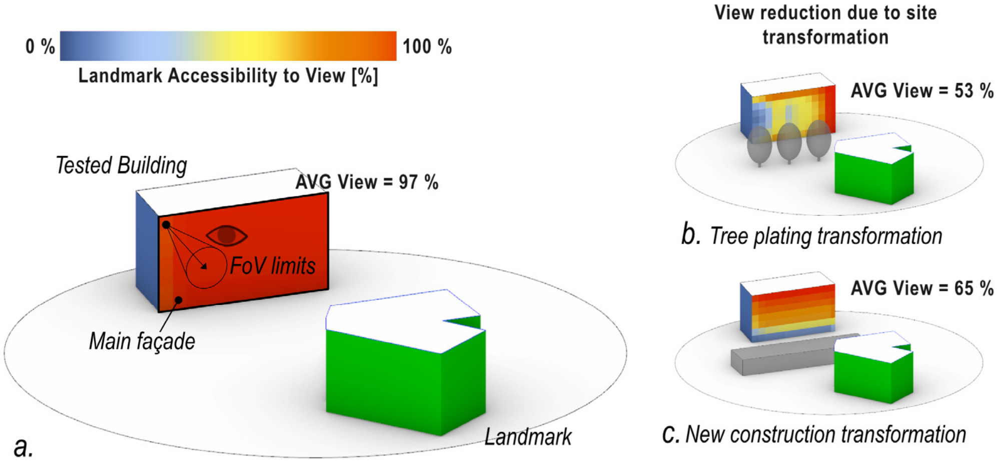

0 %
100 %
Landmark Accessibility to View [%]
View reduction due to site transformation
AVG View = 53 %
Tested Building
AVG View = 97 %
FoV limits
Main façade
Landmark
a.
b. Tree plating transformation
AVG View = 65 %
c. New construction transformation

Figure 18. The figure displays the target analysis performed within the same scene as Figure 8 with the assessment algorithm developed in this study. Three scenes are evaluated: the scene without visual obstructions (**a**), a transformation plan where a small group of trees is planned (**b**), and another one where a small structure is added in front of the main façade (**c**). The visual access to the landmark recorded by the algorithm displays the percentage of visual access for each.

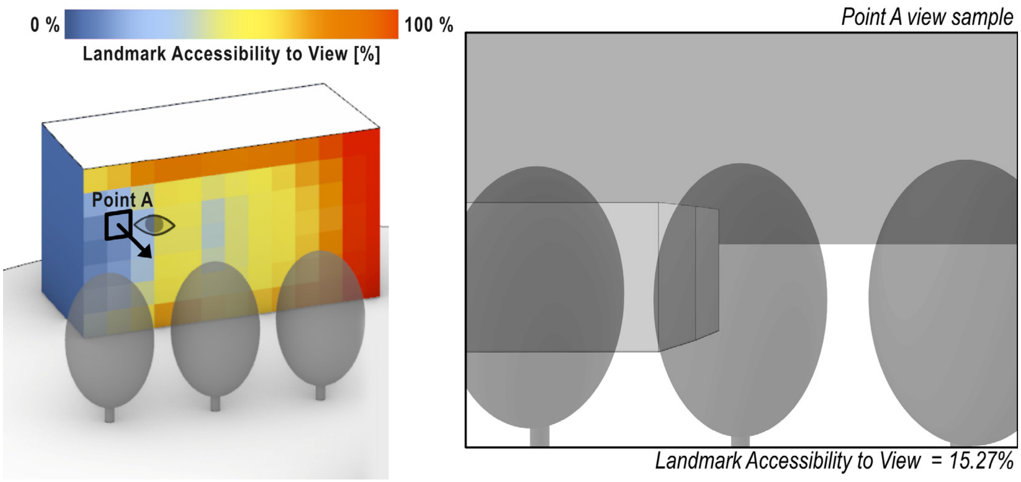

0 %
100 %
Landmark Accessibility to View [%]
Point A
Point A view sample
Landmark Accessibility to View = 15.27%

Figure 19. The figure details the influence of planting new trees in providing visual access to the outside, as seen from the analysis point A.

Figure 19 provides a more detailed display of the reduction effects of the planned obstructions. It references a point of view with remarkably low visual accessibility to the landmark to visualise the visual content incoming towards that location.

The softBIM approach streamlines data exchange between CAD and GIS environments by embedding visual analysis results directly within geometries as attribute lists. This process eliminates the need for manual data re-assignment in GIS platforms or the necessity to co-import multiple separate files and ensure the subsequent information matching. The complete automation achieved by the script developed in this study highlights the significant advantages of using softBIM approaches. These methods allow for the efficient management of not only visual analysis results but also any quantitative evaluation of the built environment. The results demonstrate the feasibility of implementing a softBIM approach to enhance automation and collaboration in complex built environment analyses within Rhinoceros, as the developed algorithm successfully computes a visual metric currently unavailable in built-in tools of reviewed software. This achievement matches the study's objective of improving visual analysis capabilities, though it is seen as a prototype—an initial step testing the viability and potential advantages of the approach. Future stages will involve more complex visual metrics and test cases to expand the study's impact and provide valuable recommendations for applying similar workflows. At this stage, a key practical recommendation emerging from the study relates to execution speed issues in complex projects, as Grasshopper inherently demands significant effort to navigate and optimize existing scripts. The complexity of revising a project can aggravate performance issues, making it crucial to address inefficiencies early on to ensure that algorithms run smoothly as the project scales. In particular, looped instructions can represent a potential bottleneck in Grasshopper script execution. While specific add-ons (e.g., Anemone [52]) can implement such operations, it is important to note that Python-based or C++-based integrations can significantly improve calculation speed [53]. In this regard, the Pancake [54] add-on's analysis function can generate detailed reports on a Grasshopper script's execution time. This type of analysis helps identify the sections of the script that most impact performance, providing valuable insights to guide potential optimization efforts. In this instance, the Pancake analysis performed on the study's algorithm proved that the most critical process is indeed the loop described in Figure 12, which cycles the calculation of each geometry in each input dataset. Table 2 displays the results, highlighting that each loop cycle takes 155 ms to compute (i.e., each point of view takes 155 ms to compute the visual analysis). This high execution time confirms Zheng et al. observations [54], as the algorithm implements the add-on Anemone to perform the loop function. Future development of the algorithm may need to develop different custom components to increase the execution performance.

Table 2. Outcome of the performance analyser derived from the script execution. The algorithm is currently able to compute the visual results of 6.5 points of view per second of execution. For reference, a building facade covering an area of 1000 m^2, divided into a sample grid with 1 m cell edges (resulting in 1000 points of view), can be fully visually analysed in approximately 2.5 min.

Whole Script (Loop is Run Only 1 Cycle)	**Execution Time**	**%**	**Iteration**
Reference by Layer (R6)	380 ms	7.6	1
Reference by Layer (R6)	92 ms	1.8	1
GenPts (LB Generate Point Grid)	43 ms	0.9	1
Filter by User Attribute (R6)	35 ms	0.7	1
Construct Point	32 ms	0.6	500
Collision One | Many	17 ms	0.3	500
Collision One | Many	15 ms	0.3	500
Other components	378 ms	87.2	Various
TOTALE	992 ms		
Body of the Loop (1 Cycle)	**Execution Time**	**%**	**Iteration**
GenPts (LB Generate Point Grid)	43 ms	20.5	1
Construct Point	32 ms	15.3	500
Collision One | Many	17 ms	8.2	500
Collision One | Many	15 ms	7.2	500
Other components	48 ms	48.8	Various
TOTALE	155 ms		

6. Conclusions

Content-based visual data is increasingly becoming an important aspect to consider in the design of the built environment. Visual analysis, which refers to procedures used to quantify and qualify the exchange of visual information between different spatial locations, can be performed using a limited number of specialized software and methodologies. This study utilized the Grasshopper Visual Programming Language to prototype a custom algorithm designed to support existing visual analysis tools within the same environment. The algorithm was developed to perform targeted visual analysis of entities, measuring the impact of visual obstructions on the visibility of specific landmarks. It generates a percentage value indicating the degree of obstruction, ranging from 100% (completely unobstructed view) to 0% (completely obstructed view). The domain range of the metric is consistent and is not dependent on the targeted object geometry. In addition, a review of visual analysis software capabilities has recognized that increased support for flexibly controlling and accounting for the impact of human field-of-view limits could represent an important issue to address. In this regard, the study's algorithm can limit the visual field of the analysis to a specific visual cone centered around a set line of sight direction. Both parameters can be freely controlled to increase the accuracy of visual analysis to simulate and output realistic assessments. In the present study context, the algorithm has been applied to sample the flux of visual information incoming toward building facades. Points of view have been mapped onto the vertical simulating visual conditions of people standing at envelope openings (i.e., windows). In addition, the gaze direction was set to be normal to the building surface and pointing outwards, while the cone of view was set to be 60° wide to limit the observation to areas with higher visual quality. Both parameters aim to increase the fidelity to the envisioned scenario of accessing the outside view. The function can be deployed to complement the assessments developed via the visibility percent analysis already available in the Grasshopper ecosystem's frame of reference. This analysis measures the visible percentage of the target object shell surface, developing an assessment whose peak value varies depending on the targeted object geometry while not filtering the analysis computation by gaze direction or field of view limit.

Finally, the study verified the value of configuring a softBIM workflow in this setting. The synergy achieved by the combined implementation of a selected number of add-on command groups enabled the generation of softBIM files capable of storing embedded data in data spaces directly accessible by querying relevant geometries in the file. The algorithm structure was functionally partitioned into individual functions, each designed to output a specific dataset populated by geometries with assigned relevant information. For example, Dataset 1 contained the building geometries composing the analysis site, where each building was assigned a label identifying its status as a landmark to analyse. Similarly, Dataset 5 contained all the cell geometry resulting from each building envelop subdivision in the analysis grid. In this instance, each cell stores the list of visual assessment results generated by the analysis. SoftBIM capabilities also enabled the coding of an automated export function to transfer visual analysis results from CAD to a GIS environment, a feat usually complex and potentially requiring manual data matching between multiple files. Similar applications may increase the interoperability of visual analysis results, with the consequence of increasing their efficacy in each design process.

However, the study highlights the inherent limitations of Grasshopper's VPL in handling algorithms that involve long instruction loops, which can slow down execution. In such cases, code optimization is crucial to ensure that a Grasshopper script can perform effectively in real-world applications within a feasible timeframe.

Code optimization, along with testing the methodology on a broader set of concrete test cases, are key objectives for the next phase of the study's development.

Author Contributions: Conceptualization, M.C., A.S., J.D.B.C., A.G.M. and T.P.; methodology, M.C.; software, M.C.; validation, M.C., T.P. and A.S.; formal analysis, M.C. and A.S.; investigation, M.C. and A.S., resources, M.C.; data curation, M.C. and T.P.; writing—original draft preparation, M.C.,

A.S. and T.P.; writing—review and editing, M.C., A.S., J.D.B.C., A.G.M. and T.P.; visualization, M.C.; supervision, A.S., J.D.B.C., A.G.M. and T.P.; funding acquisition, T.P. and A.S. All authors have read and agreed to the published version of the manuscript.

Funding: Ministry of Universities and Research (MUR), within the REACT-EU-PON framework initiative to address green transition (Green—CUP D45F21003540001) and the support of the ABC Department's Scientific Committee to fund relevant equipment for the study (Funding for research valorisation activities TM DABC2020 and TM DABC2022).

Data Availability Statement: The datasets presented in this article are not readily available because the data are part of an ongoing study. Requests to access the datasets should be directed to corresponding author.

Acknowledgments: This work has been made possible thanks to the support provided by SEED-Lab@DABC (Politecnico di Milano, ABC Dept.) on the technological support and knowledge shared.

Conflicts of Interest: The authors declare no conflict of interest.

Nomenclature

BIM	Building Information Modeling
GIS	Geographic Information System
VF	Visual field
FOV	Field of view
LOS	Line of sight
FP	Far point
NP	Near point
VA	Visual absorption
EIA	Environmental impact assessment
VIA	Visual impact assessment
ZTV	Zone of Theoretical Visibility
ZVI	Zone of Visual Influence
DTMs	Digital terrain models
DEMs	Digital elevation models
DSMs	Digital Surface Models
VI	Visual intrusion

References

1. Tzempelikos, A.; Shen, H. Comparative control strategies for roller shades with respect to daylighting and energy performance. *Build. Environ.* **2013**, *67*, 179–192. [CrossRef]
2. Lynch, K. *Managing the Sense of a Region*, 2nd ed.; The MIT Press: Cambridge, UK, 1976. Available online: https://archive.org/details/managingsenseofr0000lync_f6l5/page/n6/mode/1up?q=%22visual+absorption%22 (accessed on 10 August 2023).
3. Bittermann, M.S.; Ciftcioglu, O. Visual perception model for architectural design. *J. Des. Res.* **2008**, *7*, 35. [CrossRef]
4. Florio, P.; Scartezzini, J.-L.; Cristina, M.; Probst, M. Towards a GIS-based Multiscale Visibility Assessment Method for Solar Urban Planning. Ph.D. Thesis, École Polytechnique Fédérale De Lausanne, Vaud, Switzerland, 2018. Available online: https://www.researchgate.net/publication/328051464_Towards_a_GIS-based_Multiscale_Visibility_Assessment_Method_for_Solar_Urban_Planning (accessed on 10 August 2023).
5. Sarihan, E. Visibility Model of Tangible Heritage. Visualization of the Urban Heritage Environment with Spatial Analysis Methods. *Heritage* **2021**, *4*, 2163–2182. [CrossRef]
6. Zheng, H.; Wu, B.; Wei, H.; Yan, J.; Zhu, J. A Quantitative Method for Evaluation of Visual Privacy in Residential Environments. *Buildings* **2021**, *11*, 272. [CrossRef]
7. Abdelrahman, M.; Coates, P.; Poppelreuter, T. Visible outside view as a facilitation tool to evaluate view quality and shading systems through building openings. *J. Build. Eng.* **2023**, *80*, 108049. [CrossRef]
8. Grêt-Regamey, A.; Bishop, I.D.; Bebi, P. Predicting the scenic beauty value of mapped landscape changes in a mountainous region through the use of GIS. *Environ. Plan B Plan Des.* **2007**, *34*, 50–67. [CrossRef]
9. Cilliers, D.; Cloete, M.; Bond, A.; Retief, F.; Alberts, R.; Roos, C. A critical evaluation of visibility analysis approaches for visual impact assessment (VIA) in the context of environmental impact assessment (EIA). *Environ. Impact Assess. Rev.* **2023**, *98*, 106962. [CrossRef]
10. Zones of Theoretical Visibility (ZTV). Available online: https://www.2bconsultancy.co.uk/ztv.htm (accessed on 10 August 2023).
11. Jeffery, A. Zone of Theoretical Visibility Maps. Available online: https://www.landscapevisual.com/zone-of-theoretical-visibility-maps/ (accessed on 10 August 2023).

12. Mazzeo, A.; Arcidiacono, C.; Valenti, F.; Leonardi, M.; Porto, S.M.C. Viewshed Analysis-Based Method Integrated to Landscape Character Assessment: Application to Landscape Sustainability of Greenhouses Systems. *Sustainability* **2022**, *15*, 742. [CrossRef]
13. Benedikt, M.L. To take hold of space: Isovists and isovist fields. *Environ. Plan B Plan Des.* **1979**, *6*, 47–65. [CrossRef]
14. Tandy, C. The isovist method of landscape survey. *Methods Landsc. Anal.* **1967**, *10*, 9–10.
15. Verhoeven, G.J.; Santner, M.; Trinks, I. From 2D (to 3D) to 2.5D—Not All Gridded Digital Surfaces Are Created Equally. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2021**, *8*, 171–178. [CrossRef]
16. Ledoux, H.; Peters, R.; Ohori, K.A. What Is a Digital Terrain Model? Available online: https://3d.bk.tudelft.nl/courses/backup/geo1015/2018/data/geo1015_01.pdf (accessed on 13 May 2024).
17. Dawes, M.J.; Ostwald, M.J. Isovists: Spatio-visual Mathematics in Architecture. In *Handbook of the Mathematics of the Arts and Sciences*; Springer International Publishing: Cham, Switzerland, 2021; pp. 1419–1431. [CrossRef]
18. SpaceGroupUCL. Available online: https://github.com/SpaceGroupUCL/depthmapX/releases (accessed on 13 March 2023).
19. ArcGIS Pro. Available online: https://en.wikipedia.org/wiki/ArcGIS_Pro (accessed on 11 September 2023).
20. Visibility Analysis. Available online: https://plugins.qgis.org/plugins/ViewshedAnalysis/ (accessed on 11 September 2023).
21. DeCodingSpaces Toolbox | Computational Analysis and Generation of STREET NETWORKS. Available online: https://toolbox.decodingspaces.net/ (accessed on 11 September 2023).
22. Hawk. Available online: https://www.food4rhino.com/en/app/hawk (accessed on 11 September 2023).
23. 3.4.6 Generate and Export an Minkowski Model. Available online: https://isovists.org/user-guide/3-4-6-generate-and-export-an-minkowski-model/ (accessed on 11 September 2023).
24. VISUAL TRACER. Available online: https://www.food4rhino.com/en/app/visual-tracer (accessed on 11 September 2023).
25. Climate Studio. Available online: https://www.solemma.com/climatestudio (accessed on 11 September 2023).
26. Isovists.Org. Available online: https://isovists.org/ (accessed on 11 September 2023).
27. Ladybug Tools | Home Page. Available online: https://www.ladybug.tools/ (accessed on 11 September 2023).
28. Revit. Available online: https://www.autodesk.com/it/products/revit/overview?term=1-YEAR&tab=subscription (accessed on 27 August 2024).
29. Archicad. Available online: https://graphisoft.com/solutions/archicad/archicad-collaborate?gad_source=1&gclid=CjwKCAjw8rW2BhAgEiwAoRO5rL2VEUsJd7jH_S-C9LBu_K7Sh7jvFjMrFTlGHG0U5igqUr9ldhGKjhoC3eEQAvD_BwE (accessed on 27 August 2024).
30. Revit Daylight Model Import. Available online: https://climatestudiodocs.com/docs/revitImporter.html (accessed on 27 August 2024).
31. Nayak, N. Exploratory Visibility Analysis from Revit Building Rooms. Available online: https://www.esri.com/arcgis-blog/products/arcgis-pro/3d-gis/exploratory-visibility-analysis-from-building-rooms/ (accessed on 27 July 2024).
32. Grandjean, E. *Fitting the Task to the Man: A Textbook of Occupational Ergonomics*, 4th ed.; Taylor & Francis: London, UK, 1988. Available online: https://archive.org/details/fittingtasktoman0000gran/page/233/mode/1up?q=%22visual+field%22 (accessed on 10 August 2023).
33. Hollows, G.; James, N. Understanding Focal Length and Field of View. Available online: https://www.edmundoptics.eu/knowledge-center/application-notes/imaging/understanding-focal-length-and-field-of-view/ (accessed on 10 August 2023).
34. Focal Length of a Human Eye. Available online: https://hypertextbook.com/facts/2002/JuliaKhutoretskaya.shtml (accessed on 10 August 2023).
35. Automated Static Perimetry. *Am. Orthopt. J.* **1992**, *42*, 187. [CrossRef]
36. Stidwill, D.; Fletcher, R. *Normal Binocular Vision*; Wiley: Hoboken, NJ, USA, 2010. [CrossRef]
37. Barden, A. Binocular Vision, Eye Teaming and Binocular Vision Dysfunction. Available online: https://www.allaboutvision.com/eye-care/eye-anatomy/what-is-binocular-vision/ (accessed on 10 August 2023).
38. Cameron, J.R.; Skofronick, J.G.; Grant, R.M. *Physics of the Body*; Medical Physics Publishing: Madison, WI, USA, 2017. [CrossRef]
39. Strasburger, H.; Rentschler, I.; Juttner, M. Peripheral vision and pattern recognition: A review. *J. Vis.* **2011**, *11*, 13. [CrossRef] [PubMed]
40. ELEFRONT (Da Elevelle). Available online: https://www.food4rhino.com/en/app/elefront (accessed on 12 September 2023).
41. Fagerström, G.; Hoppermann, M.; Almeida, N.; Zangerl, M.; Rocchetti, S.; Van Berkel, B. Softbim: An Open Ended Building Information Model in Design Practice. In Proceedings of the 32nd Annual Conference of the Association for Computer Aided Design in Architecture, San Francisco, CA, USA, 18–21 October 2012; pp. 37–46. [CrossRef]
42. Introduction. Welcome to eleFront! Available online: https://docs.elefront.io/ (accessed on 12 September 2023).
43. Visibility Percent. Available online: https://docs.ladybug.tools/ladybug-primer/components/3_analyzegeometry/visibility_percent (accessed on 1 September 2022).
44. LB Visibility Percent.py. Available online: https://github.com/ladybug-tools/ladybug-grasshopper/blob/master/ladybug_grasshopper/src/LB%20Visibility%20Percent.py (accessed on 1 September 2022).
45. Llobera, M. Extending GIS-based visual analysis: The concept of visualscapes. *Int. J. Geogr. Inf. Sci.* **2003**, *17*, 25–48. [CrossRef]
46. Kutzner, T.; Chaturvedi, K.; Kolbe, T.H. CityGML 3.0: New Functions Open Up New Applications. *PFG—J. Photogramm. Remote Sens. Geoinf. Sci.* **2020**, *88*, 43–61. [CrossRef]
47. Biljecki, F.; Ledoux, H.; Stoter, J. An improved LOD specification for 3D building models. *Comput. Environ. Urban Syst.* **2016**, *59*, 25–37. [CrossRef]

48. Export Vector. Available online: https://grasshopperdocs.com/components/heron/exportVector.html (accessed on 13 November 2023).
49. Heron. Available online: https://www.food4rhino.com/en/app/heron (accessed on 13 November 2023).
50. Zlatanova, S.; Rahman, A.A.; Pilouk, M. 3D GIS:Current status and perspectives. In Proceedings of the Symposium on Geospatial Theory, Processing and Applications, Ottawa, ON, Canada, 9–12 July 2002. Available online: https://www.researchgate.net/publication/228779275_3D_GIS_Current_status_and_perspectives/citations (accessed on 20 November 2023).
51. Skeletonize. Available online: https://scikit-image.org/docs/stable/auto_examples/edges/plot_skeleton.html (accessed on 20 November 2023).
52. Zwierzycki, M. Anemone. Available online: https://www.grasshopper3d.com/group/anemone (accessed on 1 September 2022).
53. Zheng, H.; Guo, Z.; Liang, Y. Iterative Pattern Design via Decodes Python Scripts in Grasshopper. In Proceedings of the 18th CAAD Futures Conference, Daejeon, Repulic of Korea, 26–28 June 2019.
54. Pancake. Available online: https://www.food4rhino.com/en/app/pancake (accessed on 21 November 2023).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.