543

Designerly Way of Investigating Space Syntax Software

PELIN DURSUN CEBI 1, NILUFER KOZIKOGLU 2, TUGRUL YAZAR 3, & MELIKE SENA ERDEN 1,

1 ISTANBUL TECHNICAL UNIVERISTY, 2 TUSPA ARCHITECTS, 3 ISTANBUL BILGI UNIVERISTY

ABSTRACT

This study aims to problematize Space Syntax spatial analysis software by emphasising the architectural design process and to discuss how the existing toolkit can be enriched with qualities that can accompany this process. Within the scope of the study, the possibilities and limitations of existing Space Syntax computer software are examined. Existing software has been comparatively evaluated and mapped in terms of the main objectives of the programs, their operation and the spatial information they reveal. Then, the qualities of the software as a design tool were questioned. It is thought that the reading on Space Syntax software will reveal an awareness of how a scientific understanding can be translated into the design process as a design tool that will show analytical, creative, dynamic, and interactive features, what kind of qualities Space Syntax software has in this sense and how it can be developed in the future. The findings can be a guide for new software designs.

KEYWORDS

Space Syntax Software, Architectural Design, Design Thinking

1 INTRODUCTION: DESING THINKING AND SPACE SYNTAX, TRANSFORMATION FROM ANALYTIC TO CREATIVE MANNER

Architectural design is a "reflection by action" or “learning-by-doing” inquiry (Schön 1987) and exploration process (Dursun 2007) centred on problem-solving (Lawson 2003), in which developed ideas are evaluated, experimented with, and experienced repeatedly. This continuous testing/learning (Lawson and Dorst 2009) and redoing also represent a "game-like" mechanism through which new possibilities and propositions emerge. This game is based on experience, which enriches the training progress, and is run within basic rules that are open to probing and opting. The "game-like" procedure allows for an open-ended, interactive mode of Operandi for the designer, who questions, criticizes, and restructures the motions rather than simply observing them. The "game-play" also necessitates a reality check via representation. To put it another way, design decisions must be externalised in order to initiate the interactive dialogue between the designer and her/his cognitive construction. The reflective and iterative illustrations, as well as their assessment, drive the dialogue. This process necessitates the collaboration of rational, analytical, and creative thinking abilities, as well as abstraction and evaluation abilities (Lawson and Dorst 2009).

The incorporation of the scientific approach to form-finding, or morphological studies, into architectural design has been an intriguing research topic. These studies provide a language for discussing (Al Sayed 2014) and interpreting architectural formation, as well as ascribe the relationship between design and its social implications (Hanson 2001). This language enables the representation of architectural decisions as well as the exploration of the potentials of designed spatial forms. The primary focus of this research is on how scientific tools and methodologies can be used not only for analytic and explicit assessment, but also for an inspirational, informative, and productive course of design.

Space Syntax, as a morphological modality, considers spatial formations as relational constructs (Hillier 1996). Spatial experience is shaped through these relational structures that follow specific patterns and rules. The arrangement of the spaces reveals a certain interaction model among the users, directs the movements and actions of the bodies, and thus produces some social-cultural meanings. Space Syntax intends to make this non-discursive relational quality discursive with the social knowledge coded in it by analysing these relational patterns and synthesizing a correlation between the "designed" and the "perceived/lived" space (Al Sayed 2014, Psarra 2010). As a result, Space Syntax is enticing to research focused on the design process/design practice (Dursun 2007). By evaluating what has been designed, Space Syntax provides the designer with scientific data on the formation and use of spatial configurations. These may infer that the emphasis of Space Syntax is on an analytical approach. The main question in this study is to investigate how the scientific tools of Space Syntax can be used in a creative, creative, and productive manner rather than purely descriptive-analytical evaluation (Koch and Carranza 2014). In such a case, the process is almost certainly not focused on examining and testing a spatial configuration as a result product, as is frequently the case with most software and their applications. The bottom line is that a plethora of possibilities can emerge during the early stages of the design process, as well as through its evolution with a series of design decisions using an analytical, scientific (reason-based) approach in design thinking. In that case, an internal interactive and creative reflection is being carried out with the help of tools and repetitive computation. Scientific data becomes a creative ground on which the designer can think, share, and co-design for the designer, who constantly reformulates his/her thinking by doing and disrupting it in a play-like research and discovery process.

Space Syntax has created a number of tools that allow us to discuss the qualities of spatial formations. These tools are a collection of mappings, diagrammatic expressions, syntactic measures, software, and plugins that are used to represent and analyse spatial formations. Parallel to the technologies involved, a variety of software has been developed in distinct features and functionalities across multiple platforms since the 1980s, and this quest continues today.

The aim of this research is to evaluate existing software with a focus on how the design process works, and to discuss how existing syntactic toolkits can be enriched with qualities that can better accompany this process. To that end, the Space Syntax software is first compared in terms of their objectives, working principles, spatial information generated, and the various forms in which this information is represented, all of which are thought to be important in creating an understanding of these software's usability in the design process. The features they have are then used to evaluate what kind of design tool they are in the dialogue between the designer and the design, as well as what kind of thinking and making process they feed into this relationship. The main interest here is what kind of dialogues are established through these tools for the designer/architect during the actual course of design, what kind of data these dialogues produce and how they contribute to the design process.

The preceding reading on Space Syntax software as a design tool will demonstrate scientific not only analytical but also investigative, creative, dynamic, and interactive understanding of the design process, resulting in an awareness of the potentials Space Syntax software has in this regard and how they can be developed in the future. The findings can also be used to guide the development of new software.

2 EXPLORING SPACE SYNTAX SOFTWARE

It is seen that Space Syntax, together with its theoretical framework, has been the subject of many research agenda from the 1980s to the present, and the methods used to read/understand space have also developed and diversified in this process. While the abstract-graphic spatial representations required for space analysis were originally created with sketch paper and pencil, they may now be modelled with a wide range of computer programs, and spatial characteristics and measurements can be displayed both diagrammatically and numerically. These software have evolved from the simple but pioneering software -Apha Syntax- developed by Paul Coates (Coates 2010) to explore spatial stacks by building computational prototypes, into software developed by Nick Ship Dalton and Alasdair Turner, used at urban and building scale. The three-dimensional VGA studies developed by Varoudis and Psarra (2014) on the basis of Turner's studies, have been enriched by focusing on the different qualities of the space with real-time space syntax analysis, including the size of Nourian (2016).

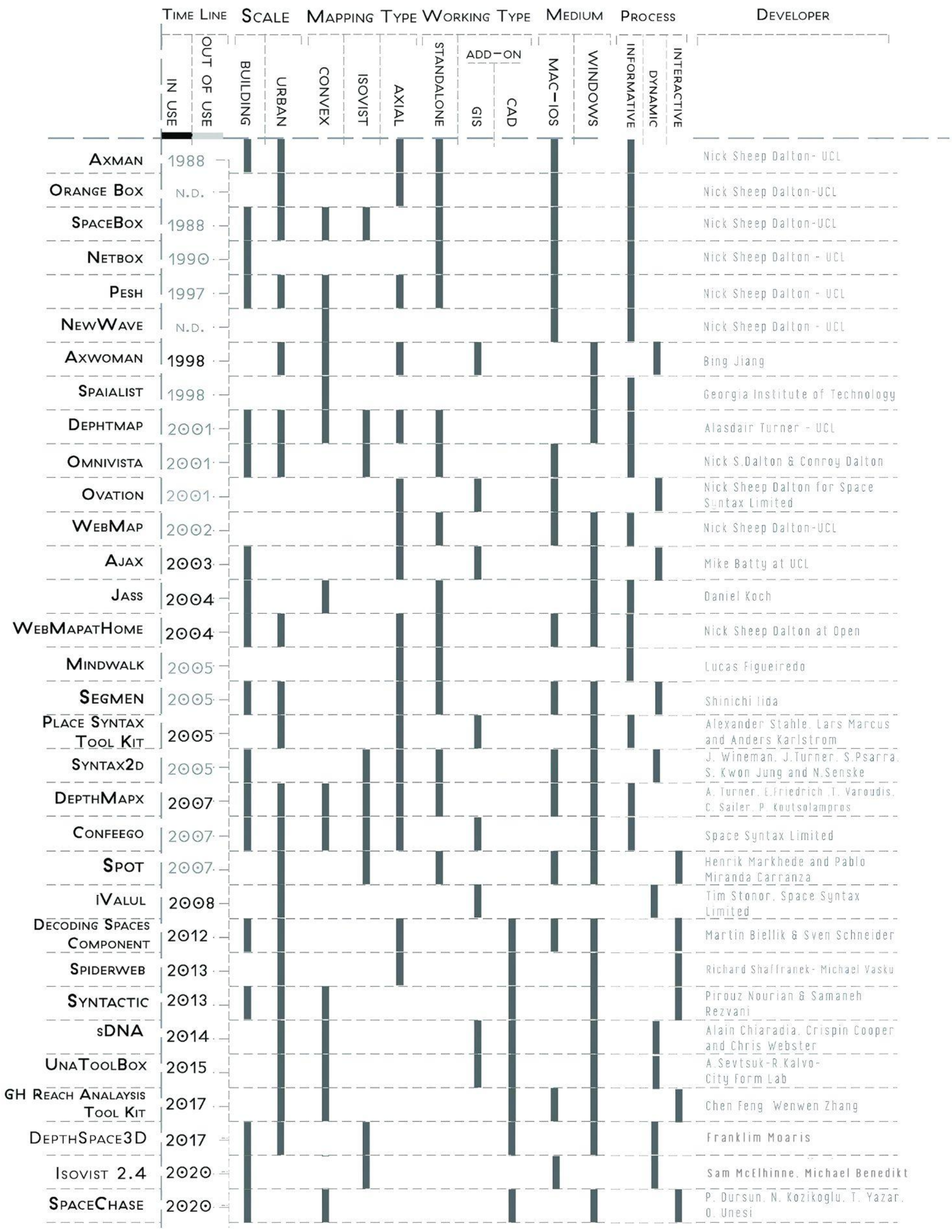

Figure 1: Classification of tools within the timeline

Through the research on the past and present software published through the Space Syntax community it has been observed that in a range of two decades there have been developed over thirty software packages. Figure 1 reveals some of these programs, but not all of them are in use. It is seen that these software diversify by using different interfaces (mac, pc) in partnership with different disciplines, computer programmers, mathematicians, architects, urban planners, keeping different mappings in the centre and focusing on one or more of them ("axial" or one-dimensional organization - movement routes, "convex" or two-dimensional organization - fields of action and interaction, and **"isovist"** - visual fields that change with movement). It can be suggested that software has different qualities, potentials, and limitations in terms of its working principles (whether there is a need for another program, whether it is open source and allows other researchers to develop it), the data sets (different maps, observation data, statistical data, measurements, etc.) it can bring together, the new metrics and qualities it tries to include in the space digitization process, (third dimension, size, context, push and pull forces), the possibilities it provides through scales (building and urban) and the outputs (diagrammatic expressions, numerical data) it produces. The diversity of these software, which builds its algorithm on Space Syntax theory and intends to use a common language over numerical measurements, is exciting considering the ongoing research. The process undoubtedly proceeds through the demands and requirements that arise in the continuous interaction of theory research and practice. It can be described as a collection of tools that are being developed and updated through various experiments, with no end in sight due to technological advancements.

Netbox and ***Jass***, for example, were among the first implementations of these programs, and they are capable of producing justified graphs of spatial systems as well as numerical data. Netbox software that ran on a Mac in the 2000s is no longer supported (Dursun 2020). Developed by Daniel Koch in 2004, Jass focuses on *convex spaces* and allows justified graphs to be created in a simple interface. The program performs Space Syntax analysis by establishing nodes that express spatial units and the relationships between them, and metrics are calculated and revealed (Koch 2004).

Spatialist (Peponis and Wineman 1997) is another software by John Peponis and Jane Wineman from Georgia Institute of Technology that is concentrated on *convex space analysis*. “The analyses are based on the identification of the boundaries of spaces in a floor plan with certain properties, in particular concerning a viewpoint moving through the floorplan” (Do and Gros 1997). Peponis and Wineman call this as “e-partition” which represents a layout in terms of constituent convex spaces that are stable concerning visual information (Peponis and Bellal 2010). The application works in the Microstation environment.

Pesh, developed by Nick Sheep Dalton, has the potential to be used in axial and convex and shape properties analysis, façade, and symmetry tests, but it is out of date today (Dalton 1997). It allows simple drawing/measurement of *convex spaces*, especially at the building scale. The only difference between this program and ***Axman*** (Dalton 1997), which creates graphics related to the configurational structures of *axial lines*, is that it can be used to create all kinds of graphic objects (line, square, polygon, or circle) instead of simple lines to create the nodal point of the graph.

Just like Axman, which works only in Mac OS X, software such as ***Segment*** (Lida 2007)***,*** which work in both Mac OS X and Windows, ***Webmap*** (Dalton, 2002)***, WebmapAtHome*** (Dalton 2004)***, Mindwalk*** (Lucas Figueiredo, 2002)***, Place Syntax Tool*** (Ståhle et.al 2007) work on “*axial lines*” defined as “the longest straight line representing the maximum extension of a point in space” (Turner et.al 2005) in Space Syntax theory and “segment maps” produced through them. While Webmap and WebmapAtHome, which are compatible with second-generation axis mapping tools, particularly Axman, are developed as third-generation axis mapping software that goes beyond and

WebmapAtHome distinguishes itself from other axial and segmental analysis programs by being largely independent (Dalton 2007).

Depthmap (Turner 2007) developed by Alasdair Turner in 2001, ***DepthmapX*** with its updated form, and a group of software including programs such as ***Omnivista*** (Dalton and Dalton 2001), ***SPOT*** (Markhede and Carranza 2007), are all based on *isovists*, which are defined as a set of all points in space that can be seen from a certain vantage point and with a reference to a certain environment in Space Syntax theory (Benedikt 1979).

DepthmapX is an open-source software platform capable of performing a series of Space Syntax analyses in order to better understand social processes in the built environment. It works on a wide range of scales, from individual buildings to entire cities. The software's goal at any scale is to create a map of the elements of open space, connect them through a relationship (for example, with visibility [intervisibility] or overlap), and then analyse the resulting network (URL-1). ***Omnivista*** (Dalton and Dalton 2001) introduces new measures for syntactic analysis in the form of VGA analysis, including "drift" and "restricted field of view" paths.

The Spatial Positioning tool (SPOT), developed by Markhede and Carranza, aims to gain a productive character as *an interactive sketching tool* rather than just an analysis tool. According to Markhede ve Carranza "regular space syntax tools (such as Depthmap) create graphs of the occupiable space; however, the Spatial Positioning Tool (SPOT) creates graphs of the occupied. Therefore, SPOT is not strictly a tool for spatial analysis; it analyses how organisational entities occupy space concerning each other." (Markhede and Carranza 2007). SPOT has a modular and user-friendly interface that works in Windows, Mac, and Unix environments. "SPOT creates a possibility to examine the dynamics between the occupied space and the occupiable space as well as dynamics between organisational relations and spatial relations." (Markhede and Carranza 2007).

Place Syntax Tool (PST), developed by Alexander Ståhle and his team, is a plugin application for the desktop software MapInfo (Ståhle et.al 2007). The program tries to extend the axial analysis by including axial steps as part of the wider investigation of accessibility. According to Turner (2007) "The Place Syntax tool, like Markhede and Miranda's SPOT tool, turns around how we conceive space syntax to the accessibility of the occupied space, and its relationship to other spaces and other social factors contained within those spaces."

When the Space Syntax Archives are scanned retrospectively, the following programs are also encountered (URL-2): The ***Infinity Within*** software generates axial all line maps from boundary descriptions. Orange Box is a tool for super-fast axial map processing. ***NewWave*** performs integration analysis on a numerical text file. ***In Hardwave***, "integration is performed from street centrelines where space is defined as any street centreline with a name." This works in conjunction with a GIS system (Arc Info) to process integration analysis for streets by name. ***NextPesh*** is the more advanced version of Pesh that can handle curved shapes + layers + one-way streets and no right turns at junctions. ***The Urban Machine*** is a program that could allow for the interactive analysis/characterization of hundreds of different cities at the same time. ***Loglady*** is an Unix based super processor of axial maps. ***FarmerBrown*** is a Transputer based processing of integration via a vastly parallel supercomputer. ***Spacebox, which*** is developed by Dalton (1988) is an automatic generation of convex spaces and all line axial maps from a boundary/wall description.

The software analyses show that some of the software works independently, while others work as add-ons to other software. ***Axman, WebMapatHome, Depthmap, NetBox, Pesh, Omnivista, NewWave, Space Box and Orange Box, Mindwalk, Jass, Segmen, and SPOT*** are examples of independent computer software. Programs such as ***Axwoman*** developed by Bing Jiang in 1998, ***Ovation*** developed by Nick Dalton in 2001, ***Ajax*** developed by Michael Batty in 2003, ***Place Syntax Tool*** developed by Alexander Ståhle and his team, and ***Confeego*** and ***iVALUL*** (Gil et.al 2015) developed by Space Syntax Limited Company on the other hand, only work as *Geographic Information Systems (GIS) extensions*. Some programs, such as ***Spatialist***, developed by John Peponis's research group at the Georgia Institute of Technology, ***Sytanx2D*** developed by the University of Michigan, and ***Pangea***, work only as *Computer-Aided Design (CAD) extensions*. ***sDNA*** developed by Alain Chiaradia and his colleagues (Cooper and Chiaradia 2015) and ***UNA Toolbox*** developed by Sevtsuk and Mekonnen (Sevtsuk and Mekonnen 2012) can work in harmony with both CAD and GIS environments. ***ArchiSpace*** is an attempt to incorporate space syntax calculations into BIM modelling in order to incorporate space syntax into the architectural design process (Li et.al 2009).

If we take a closer look at these programs, we can uncover the following information: ***Axwoman*** is a freeware software tool for analysing the topology and scaling of very large urban street networks. It is an ArcGIS plugin (Jiang 2019). Users can perform various space partitions with drawing tools and represent the space using polygon, line, and point-based approaches. “The program calculates various measures from the Axial, Point or Housing map including connectivity, control value, integration, local integration, depth, etc., and computed results are stored in a table corresponding to the respective map theme. Users can explore data from different perspectives, or import observed data, e.g., pedestrian or other traffic flow rates for cross-analysis.” (Jiang 1988). ***Ajax,*** Analysis of Junctions and Axial lines, is a very fast open-source software that allows end-users with little knowledge of graph theory to compute measures of accessibility between streets (the primal) or measures of accessibility at junctions (the dual) (Batty 2005). The program allows the researcher to draw a series of axial lines on an input raster image/map and then determine the accessibility of streets and junctions (both to each other) (Url-3). ***Space Syntax Toolkit*** (Gil et.al 2015) is a QGIS plug-in for spatial network and statistical analysis. It acts as a front-end for the DepthmapX software within QGIS, allowing for user-friendly space syntax analysis workflows in a GIS environment. Its primary goal is to support and enhance the space syntax methodology with GIS data, analysis, and visualization features. ***Confeego*** is a tool developed by Space Syntax Limited to understand the effects of spatial configuration in urban systems or complex buildings. “It covers a range of tools that are useful for projects involving the classic space syntax analysis related to *topological depth* and other tasks related to data collection, *statistical analysis, and visualisation*. It also allows the integration of the results from other space syntax software tools. ***Confeego*** is developed as an extension to MapInfo Professional *GIS* (Geographic Information System) for Microsoft Windows and it requires an installed version of MapInfo 7.8 or above to run." (Gil et.al 2007). ***I-Valul*** is a GIS tool kit for the *socio-economic valuation of urban areas*, for designers and decision-makers (Chiaradia et.al 2008). The program, which operates as a collection of MapInfo Professional GIS platforms, aims to automate and standardize the various steps of the layout evaluation process.

Pangea is a software developed to enable the flexible analysis of three-dimensional worlds composed of objects and the spaces between them. Pangea provides a customizable tool for the examination of 3-D spatial relations by building a simple application in which 3-D shapes may be created and edited. The goal is to give the user as much flexibility as possible in terms of modifying and even creating entirely new types of analysis without having to rewrite - or even edit and recompile - the source code. Penn and his colleagues refer to the software as a "tool for thinking with," and it is used to analyse three-dimensional shape and space (Penn et.al 1997). The Pangea workbench is a simple programmable 3D CAD application. The concept of a layer has been replaced by that of a collection, which behaves similarly to a mathematical set. A simple analytic instrument known as the 'Isovist Camera' was developed in Pangea specifically for analysing visual fields in three-dimensional space (Penn et.al 1997). ***Syntax2D***, a different example from this group, is open-source software for spatial analysis that takes a critical stance toward the inward nature of space syntax software. "This software currently features a robust interface, combining existing measures such as isovist, graph, and axial analysis with newer features for path analysis." (Wineman et.al 2007).

DepthSpace3D, developed by Franklim Morais and his team, is a digital tool that works in a CAD environment, for space syntax analysis in 3D spaces. The key highlight features of the DepthSpace3D digital tool, which is working on architectural and urban studies, are transparency, multiple paths of visibility and attribution of properties to the viewed space (Morais et.al 2018). The main motivation of DepthSpace3D is to overcome the difficulty of analyzing a wide variety of dimensions and dynamic volume geometries with existing 2D Space Syntax tools.

Isovist_App, developed by Sam McElhinney and his team. The Isovist_App may display isovists in their natural state, compute inherent geometrical or relational features as measures, and display spatial representations of how such measures are dispersed in space as 'fields.' This digital tool can perform real-time point isovist, path isovist, area isovist, and isovist agent analysis and rapid field analysis (McElhinney 2020). It also comes with an integrated scatter graph tool for examining relationships (or lack thereof) between the enormous data sets generated for each measure field. This digital tool's features allow the user to investigate and isolate spatial transformations or configurative properties that have a significant impact on understanding architecture.

Spatial Design Network Analysis (sDNA) is a toolbox for 2-D and 3-D analysis, particularly street/path/urban network analysis. The program runs as an ArcGIS/QGIS/AutoCAD plugin and computes accessibility (reach, mean distance/closeness centrality, gravity), flows (bidirectional betweenness centrality), efficiency (circuity), and convex hull properties (Cooper and Chiaradia 2020).

It can be used to predict pedestrian, cyclist, vehicle, and metro flows, as well as quantify the built environment for design purposes. ***UNA, or Urban Network Analysis***, is an open-source ArcGIS toolbox developed at MIT's City Form Research Group. The toolbox can compute five different types of network centrality measures on spatial networks: reach, gravity index, betweenness, closeness, and straightness. "Although primarily developed for the analysis of urban street- and building-networks, the tools are equally suited for other spatial networks, such as railway networks, highway networks, or utility networks. Unlike previous network centrality tools that operate with two network elements (nodes and edges), the UNA tools include a third network element – buildings – that can be used as the spatial units of analysis for all measures." (Sevtsuk and Mekonnen 2017).

Furthermore, over the last ten years, various plug-ins and toolkits for design tools like Grasshopper have been developed, allowing Space Syntax analysis to be integrated into CAD environments that allow three-dimensional design and become a part of the generative design process. Among them, plug-ins such as ***Spiderweb*** (Schaffranek and Vasku 2013) and ***Decoding Spaces Components*** (Bielik et.al 2012), which work with axial lines and segment maps, are mainly used in the field of urban design and related Space Syntax analysis to the production of architectural form or urban settlement. ***Grasshopper Reach Analysis Toolkit*** (Feng and Zhang 2017) is another Grasshopper plugin that integrates Space Syntax reach analysis into the design formulation process.

Decoding Spaces Components is a graph-based spatial analysis toolset developed as an extension for Grasshopper3D for Rhino by Bielik, Schneider, and König (2012). The program's goal is to assist the designer in iteratively experimenting and making decisions with numerous variables by attempting to integrate spatial analysis into urban parametric modelling. ***Grasshopper Reach Analysis Toolkit***, which emphasizes the concept of generative design and parametric modeling in architectural design, consists of a series of Grasshopper definitions that not only enable an interactive parametric syntactic analysis but also facilitate communication with other user-defined Grasshopper components to do computational design (Feng and Zhang 2017). The tool is intended to supplement the original version of Spatialist-Lines, a GIS plugin created by Zongyu Zhang at Georgia Tech, and some measurements such as metric reach, directional reach, and directional distance per length are attempted to be incorporated into the software's structure (Feng and Zhang 2017). ***Spiderweb*** also makes an attempt to incorporate Space Syntax into parametric modelling processes. SpiderWeb, a Grasshopper plugin, intends to extend DepthmapX to another form of modeling and is applicable for three-dimensional urban and architectural design (Schaffranek and Vasku 2013). Unlike DepthmapX's well-defined metrics, Spiderweb allows different metrics to be adapted into models, and as a space quality, they can guide space research processes/decisions in the early design phase.

Unlike other tools, **Syntactic** (Nourian et.al 2013), which was developed with the inspiration of SpiderWeb (Schaffranek and Vasku 2013) plugin and works in harmony with it, is valuable in this study due to its effort to derive possible topological planar settlements from abstract spatial relationships. Syntactic, developed by Pirouz Nourian and Samaneh Rezvani of Delft University, proposes a toolkit that runs on Rhinoceros and Grasshopper (Nourian et.al 2013). Syntactic enables the creation of interactive bubble diagrams for convex spaces and aims to create diagrammatic plan layouts while presenting Space Syntax analysis in real-time (Nourian et.al 2013). The program's architectural design process begins with the creation of abstract forms/relationships. The program then generates interactive bubble diagrams with space sizes included; geographical aspects are taken into account, and the program reveals the plan schemes formed at the end of this process. Simultaneously, the program generates justified graphs by calculating the depth of the spatial network; and it computes Space Syntax metrics such as integration, control, choice, and entropy in real-time. The interactive interface of Syntactic generates plan layouts/patterns by evaluating the side-by-side, neighbourhood, and connectivity relations observed in the resulting schemas (Nourian et.al 2013).

Space Chase is a new plug-in for the Rhinoceros/Grasshopper that highlights the creative, dynamic, and interactive components of the design process (Dursun Cebi et.al 2021, Kozikoglu et.al 2020). The plug-in performs analytical Space Syntax calculations and aims for user engagement via dynamic network structure. Instead of ready-made diagrams, it provides ready-to-use outputs (such as point, line, regular, and sequential data) in the CAD environment, allowing the user to be creative during the design process.. Furthermore, the plug-in enables simultaneous design and analysis of the space and its environment by including contextual data of the intended configuration into the research process. Each space in the configuration activates, repels, and/or pulls each other in terms of its syntactic values (for example, depth and integration). Furthermore, the dynamic network, which allows designers to collaborate during the design process, groups spatial units that are likely to cluster and visually displays potential juxtapositions to the user.

3 EVALUATING SPACE SYNTAX SOFTWARE IN DESIGN THINKING AND MAKING: HOW DO THEY EVOLVE?

Based on an examination of the software's general characteristics, it is possible to conclude that these software evolve by focusing on various spatial information with different working principles and presenting this information with various visual and numerical tools. The main concern here is how this spatial information interacts with the designer during the design process, as well as how these tools communicate with the designer, thereby feeding the learning-by-doing process. While program interfaces with CAD and GIS extensions, for example, speed up spatial research, the researcher who encounters multimedia may feel alienated. As Gil (2015) and his colleagues proposed, it is critical to develop support tools that integrate space syntax's entire workflows into a single platform that is intuitive for beginners, flexible for researchers, operational for practitioners, and accessible to all users in order to improve the contribution of space syntax to design research, education, and practice.

In this regard, the study assesses the versatility of these tools in the creative design process from the perspective of the designer. The following punchlines consider the evaluation criteria:

- concerning the process: how can these tools feed the research on space (design itself)?
- concerning the tools: how can they promote dialogue between the designer and the design?
- concerning the designer: how can we discuss the role of the designer in the design process?
- concerning the design: how can we discuss the qualities of the spatial outcome?

Discussions can be enhanced on the following questions:

How might these technologies be used to enhance spatial investigation and exploration during the design process? How can these tools be deemed stimulating not only at the end of the design process, but also at the beginning of this creative process? What kind of contributions might these diagrammatic and mathematical tools make when design thinking is externalised, presented in a visible form for discussion and evaluation? What if these diagrammatic mappings were conceived of as a long-term research endeavour rather than a static visualisation? How can these tools provide a creative canvas on which the designer is constantly sketching? With the help of these tools, how can the rules and relations related to the space be interpreted and manipulated instantly in the process? How can the spatial forms that represent the body's actions/encounters/experiences be explored by exploring and increasing possibilities with this ongoing repeated exercise? What if these tools were used to explain the design process rather than the final product? How might these tools demonstrate features that are not only descriptive but also open to manipulations, restructuring, constantly communicating with the designer? How may the designer transition from "knower/interpreter" to "seeker" status during this process? How can a design product progress from a state determined by the designer's decisions in a scientific evidence-based research/discovery process to a mode in which the designer matures in action using his or her own "kinetic perception"? (Derix 2014) (Figure 2).

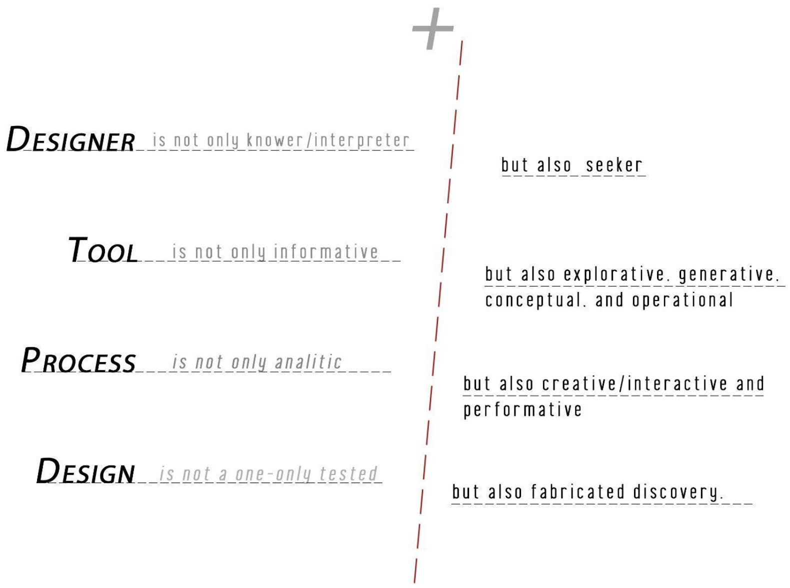

Figure 2: Changing characteristics of a design process.

Readings on Space Syntax Software reveal that the majority of the pioneering software in this field has an analytical and descriptive nature, focusing on the decoding of existing spatial formations and attempting to resolve the meaning of the configurational structure, as in the early space syntax researches (Figure 3). They are the software that are used to help determine design problems by embodying the distinctive (context/culture) features of existing spatial formations. In this case, they aimed to improve the designer's awareness and understanding of the space. Netbox, Pesh, Spacebox, Orangebox, Axman, and subsequent Webmap and DepthMap, for example, emphasize the analytical and descriptive aspects of space syntax.

On the other hand, software (sDNA, Pangea, i-Valul, UNA, etc.) that has evolved technologically over time with the support of CAD and GIS systems has gained a more *dynamic* character to present potential results to the designer during the design process and create data for the design evaluation. As the data guides new designs or revisions, the spatial-social knowledge of the design opens the process to change, transformation, and the search for new possibilities. As a result, these software actively operate on the product that emerges from the design process, but the process is typically linear. The design is finalised, and at the end of the design process, it is tested with space syntax tools and metrics. As a result, it's reasonable to conclude that the software's interaction with the user (architect) is still one-way.

A new generation of *interactive* software is being developed, allowing mutual communication about configurational concepts between the designer and the computer (Nourian et.al 2013, Dursun et.al 2021, Kozikoglu 2020). These are the tools that enable us to collaborate on thinking and designing by incorporating Space Syntax computations into the design process. As a result, the software-user relationship is bidirectional, and the relationship between the design process and the Space Syntax analysis evolves in such a way that it feeds back on itself constantly. Every design change has an effect on the Space Syntax calculations, and the change in analytical data is fed back into the design process as data. Spiderweb, Decoding Spaces Components, Syntactic, Grasshopper Reach Analysis Toolkit, and SPOT are examples of software that work as Grasshopper plug-ins. These software, which are part of research aimed at integrating Space Syntax into parametric modelling processes, combining productive and analytical models, and employing an experience-oriented (Derix 2014, Derix and Izaki 2013) design methodology, have the potential to provide designers with mind-opening tools for collaborative design. These software that allow to "co-design" from the initial stages of the design process and establish a platform whereby the result is discovered and fabricated during the progress of seeking and repose, is valuable because they reinforce the architects' mode of thinking (Lawson 2005) and promote learning from doing (Schön 1987), where the solution and problem definitions evolve together (Dorst 2010). The process is an exploratory latent with potentials (Dursun and Kozikoglu 2017). Design thinking necessitates that space syntax software evolve from a purely analytical mode to an iterative analyse-fabricate-repeat mode, allowing them to become more integrated into creative processes and interact continuously with the designer. As a result, in the future, it is critical that software progress from merely analytical to productive, with a quality that can be easily integrated into the design process and communicate with the designer.

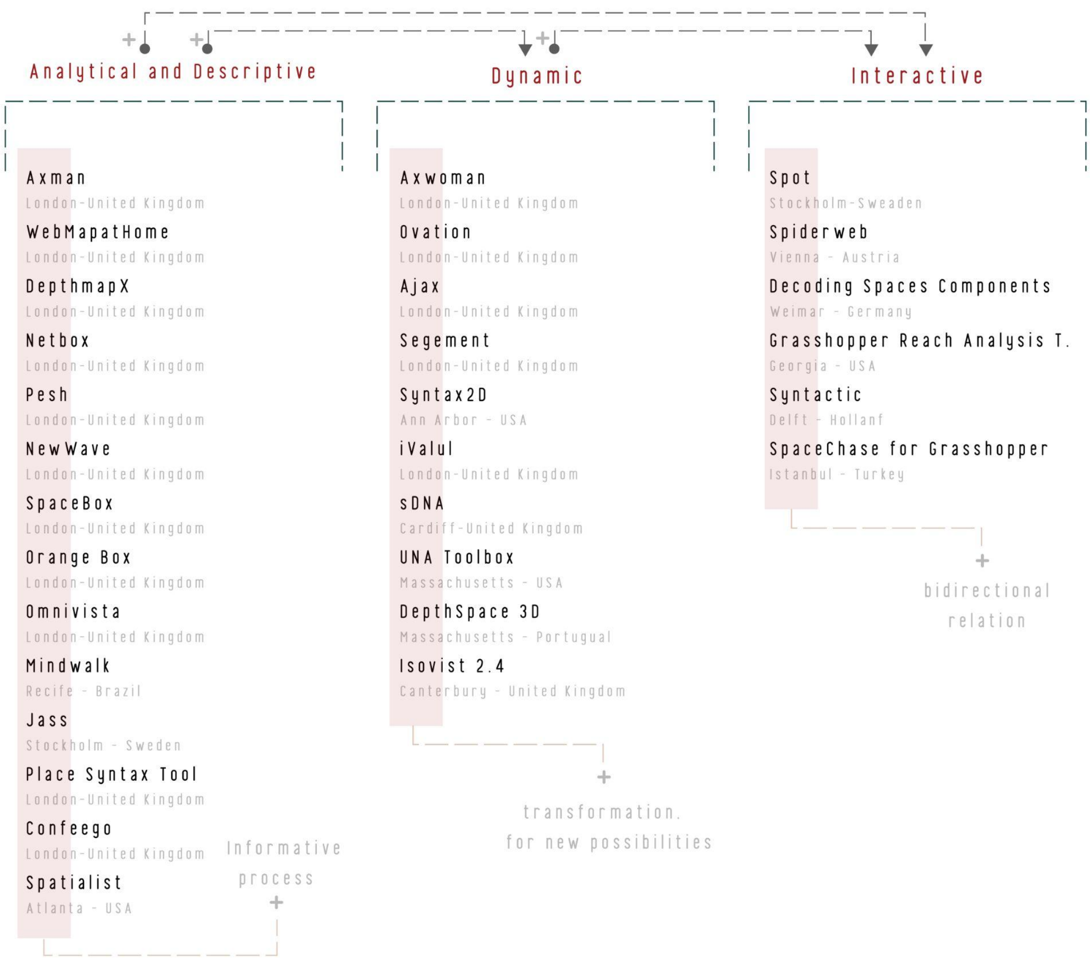

Figure 3: Different Forms of Software as Design Tool

4 CONCLUSIONS

The comparative study suggests that Space Syntax software could be used in the design process to not only provide designers with information about how the spatial interconnection in their design scheme might work at the end of the process but also to assist them in making their initial design concepts perform and become apparent. The fundamental strength of these applications as design tools, according to the designer, is their ability to generate strong links between the analytical and creative, as well as the productive.

The investigations into current Space Syntax software have led to the conclusion that software development is an ongoing pursuit by various researchers, and that the primary focus of future software research will be on how to transfer a scientific understanding into the design process using a tool with analytical, creative, dynamic, and interactive capabilities. The following principles are thought to be useful for future research:

- What a designer seeks isn't just a set of tools to aid in the process of automatic form generation, nor is it just a set of tools to aid in the analysis of fiction generated at the end of the design process. The goal is to find a set of tools that can integrate these aspects, provide real-time information to the architect, and assist in the creation of new ideas. This quality will transform the space syntax software into a diagrammatic, scientific, and creative design tool, allowing the designer to establish communication from the beginning of the design process and to explain design decisions as they are made.
- The architect is considered as the primary actor in the design process, and this is not a passive procedure in which the architect selects from a possible cloud of forms. The goal is to create a design process in which the architect actively participates, thinks about scientific data and designs, and manages them through restructuring, using a collection of tools that display interactive features. This situation necessitates the inclusion of user-friendly features in space syntax software interfaces that allow for easy communication with the architect. This property will transform a user of space syntax software into a "space creator" who learns from what he or she does rather than a "observer" who attempts to comprehend what is presented to her or him.
- The tools in question are considered in a dynamic and interactive setting, in a game-like structure in which the designer can personally participate. The goal of the game-like research process is to demonstrate that the scientific tools in question have a feature that allows the designer to think together, or to feed the intuitions with scientific data.
- Considering the design process's complex and multi-variable structure, it's critical that the tools in question progress through parametric and algorithmic processes, enriching them with various spatial qualities and expanding the realm of possibilities by allowing different relationships between variables. This feature will transform space syntax software into a sketch-like canvas where expected spatialities can be examined.

ACKNOWLEDGMENTS

The study is based on TUBITAK's ARDEB 1001 scientific research project number 119M082 on Space Syntax Design Application. Project has been run By Pelin Dursun Çebi, Nilüfer Kozikoğlu, Tuğrul Yazar, OğulCan Üneşi, Melike Sena Erden and Büşra Balaban.

Spacechase (www.spacechase.app) research and gamelike software development is currently run by Pelin Dursun Çebi, Nilüfer Kozikoğlu, Tuğrul Yazar, OğulCan Üneşi, Melike Sena Erden, Ceylin Öz and Çağla Yılmaz.

REFERENCES

Al Sayed K. (2014) 'How Design Evolve', *Journal of Space Syntax,* Vol. 5, 24 August 2014, pp.68-97, ISSN: 2044 7507.

Batty, M. (2005) 'The Ajax Project: New Theory and New Software for Space Syntax', In: *Proceedings of the 5th International Space Syntax Symposium*, Delft, The Netherlands, 13-17 June 2005, pp.43-44.

Benedikt, M. L. (1979) 'To Take Hold of Space: Isovists and Isovist Fields', *Environment and Planning B: Planning and Design,* 6(1), pp.47-65.

Chiaradia, A., Schwander C., Gil J, Friedrich E., (2008) 'Mapping the Intangible Value of Urban Layout (i-VALUL): Developing a Tool Kit for the Socio-economic Valuation of Urban area, for Designers and Decision Makers', University of Technology Eindhoven, H.J.P. Timmermans, B. de Vries (eds.), *Design & Decision Support Systems in Architecture and Urban Planning*, ISBN 978-90-6814-173-3.

Coates, P. (2010) *Programming Architecture*. London: Routledge.

Cooper, C. and Chiaradia, A. (2015) 'sDNA: How and Why we Reinvented Spatial Network Analysis for Health, Economics and Active Modes of Transport', In: *Proceeding of GIS Research UK (GISRUK) 2015,* 22 July 2015. pp.122-127.

Cooper, C.H.V, Chiaradia, A.J.F. (2020) 'sDNA: 3-d Spatial Network Analysis for GIS, CAD, Command Line & Python', *SoftwareX*, Vol. 12.

Dalton, N. (2007) WebmapAtHome, In: *New Developments in Space Syntax Software, ed. A. Turner, 6th International Space Syntax Symposium,* Istanbul, Turkey, 12-15 June 2007, pp. 7-14.

Dalton C., Dalton, N. (2001) 'Omnivista: An Application for Isovist Field and Path Analysis', In: *Proceedings of the 3rd International Space Syntax Symposium*, *Georgia Institute of Technology*, Peponis, J., Wineman, J., Bafna, S. (eds.) Atlanta, Georgia, 7-11 May 2001, pp. 25.1-25.10.

Derix, C., Izaki, Å. (2013) 'Spatial Computing for the New Organic', *Architectural Design,* 83(2), 12 March 2013, pp.42-47.

Derix, C. (2014) 'The Space of People in Computation', in Derix, C and Izaki, A (eds.) 2014, *Empathic Space: The Computation of Human-centric Architecture,* Architectural Design, John Wiley & Sons.

Dorts, K. (2010) 'The Nature of Design Thinking', In: *Proceeding of Design Thinking Research Symposium, Faculty of Design, Architecture and Building, University of Technology,* (Dorst, K., Stewart, S., Staudinger, I., Patn, B., Dong, A. (eds.), Sydney, Australia, pp. 131-139.

Do, E.Y., Gross M.D. (1997) 'Tools for Visual and Spatial Analysis of CAD Models', *Computer Assisted Architectural Design Futures* '97, R. Junge (eds.), Kluwer Academic Publishers, Dordrecht, pp. 189-202.

Dursun, P. (2002) *Morphological Analysis of Built Environment in Trabzon*. PhD Thesis, Istanbul Technical University, Istanbul, Turkey.

Dursun, P. (2007) 'Space Syntax in Architectural Design', In: *Proceedings of 6th International Space Syntax Symposium*. ITU Faculty of Architecture, Istanbul, 12-15 June 2007, pp.056.1-056.12.

Dursun Çebi, P., Kozikoğlu, N., Yazar, T., Balaban, B., Erden, M.S., Üneşi, C. (2021) *Space Syntax Design Application*, Project Report, TÜBİTAK (The Scientific and Technological Research Council of Turkey) Ardeb 1001, Project No:119M82.

Feng C., Zhang W. (2017) 'Grasshopper Reach Analysis Toolkit: Interactive Parametric Syntactic Analysis', In: *Proceedings of the 11th International Space Syntax Symposium*, Lisbon, Portugal, 3-7 July 2017, pp.159.1-159.12.

Gil, A., Stutz, J., Chiaradia, A. J. (2007) 'Confeego: Tool Set for Spatial Configuration Studies', In*: New Developments in Space Syntax Software, ed. A. Turner, 6th International Space Syntax Symposium,* Istanbul, Turkey, 12-15 June 2007, pp. 15-22.

Gil, J., Varoudis T., Karimi K., and Penn A. (2015) 'The Space Syntax Toolkit: Integrating DepthmapX and Exploratory Spatial Analysis Workflows in QGIS', In: *Proceedings of the 10th International Space Syntax Symposium*. ed. K. Karimi, L. Vaughan, K. Sailer, G. Palaiologou, and T. Bolton. Space Syntax Laboratory, The Bartlett School of Architecture, UCL (University College London): London, UK. 2015, pp.148.1-148-12.

Hanson J. (2001) 'Morphology in Design', In: *Proceedings of 3rd International Space Syntax Symposium*, Atlanta, 7-11 May 2001, pp.06.1-06.18.

Hillier, B. (1996) *Space is the Machine.* Cambridge: Cambridge University Press.

Hillier, B., Hanson, J. (1997) 'The Reasoning Art: Or, The Need for an Analytical Theory of Architecture', In: *Proceedings of 1st International Space Syntax Symposium*, Vol.1, UCL, London, April 1997, pp.01.1-01-5.

Jiang, B. (1998) Software Manual for Axwoman 3.0, 'Axwoman 3.0: An ArcView Extension for Urban Morphological Analysis' Viewed 15th January 2022 <https://www.researchgate.net/publication/237903674_Axwoman_30_An_ArcView_Extension_for_Urban_Morphological_Analysis>

Jiang, B. (2019) 'Axwoman in a Nutshell', *Technical Report*, December 2019, Viewed 15th January 2022

<https://www.researchgate.net/publication/337656401_Axwoman_in_a_Nutshell>

Koch, D. (2004) *Spatial Systems as Producers of Meaning: The idea of Knowledge in Three Public Libraries*. Licentiate Thesis, Stockholm: KTH School of Architecture/Spatial Analysis

Koch, D. (2010) 'Spatial Positioning Tool: A Prototype Software and Some Background Correlation Data'. In: *Proceedings of 6th International Space Syntax Symposium,* Vol. 2, Istanbul, 12-15 June 2007, pp. 69.1-69.14.

Koch, D., Miranda Carranza, P. (2014) 'Models and Diagrams in Architectural Design', *Journal of Space Syntax,* 5(1).

Kozikoğlu, N., Dursun Çebi, P., Yazar, T., Balaban, B., Üneşi, C., Erden, M.S. (2020) 'Dynamic Architectural Canvas: Designing a Relational Mapping Based Architectural Design Tool'. In: *The Proceedings of eCAADe 2020: Anthropologic- Architecture and Fabrication in the Cognitive Age*, Vol. 1, Berlin, 16-17 September 2020, pp.229-238.

Lawson, B. (2003) *How Designers Think*. Oxford: Elsevier.

Lawson, B., Dorst, K. (2009) *Design Expertise*. Oxford: Elsevier.

McElhinney, S. (2020), 'The Isovist_App: A Basic User Guide', version 1.6, Viewed 10th March 2022

<https://isovists.org/user-guide/1-1-what-is-the-isovist_app/>

Morais F. (2018) 'DepthSpace 3D: A new Digital Tool For 3D SpaceSyntax Analysis' David Leite Viana, Franklim Morais, Jorge Vieira Vaz (eds.), *Formal Methods in Architecture and Urbanism*, Cambridge Scholars Publishing; 1st edition (April 1, 2018) pp.91-111, ISBN-10:1527507629.

Nourian, P., Rezvani, S., Sarıyıldız, S. (2013) 'Designing with Space Syntax, A Configurative Approach to Architectural Layout, Proposing a Computational Methodology', *Computation and*

Performance, In: *Proceeding of 31st eCAADe Conference*, Vol. 1, Delft, the Netherland, 18-20 September 2013, pp.357-365.

Nourian P., Rezvani, S. and Sariyildiz, I.S. (2013). 'A Syntactic Design Methodology: Integrating Real-time Space Syntax Analysis in a Configurative Architectural Design Process', In: *Proceedings of the 9th International Space Syntax Symposium,* Seoul, 13 December 2013, pp.048.1-048.15.

Nourian, P. (2016) *Configraphics: Graph Theoretical Methods for Design and Analysis of Spatial Configurations*. PhD thesis. Delft University of Technology.

Penn, A., Conroy-Dalton, R., Dalton, N., Dekker, L., Mottram, C., Turner, A. (1997) 'Intelligent Architecture: New Tools for the Three Dimensional Analysis of Space and Built Form', In: *First International Space Syntax Symposium,* Vol. 2, London, April 1997, pp.30.1-30.19.

Peponis, J., Wineman, J. (1997) *Spatialist-Partitions*, Computer program for Microstation Software, Copyright Georgia Tech Research Corporation.

Peponis, J., Bellal, T. (2010) 'Fallingwater: The Interplay Between Space and Shape', *Environment and Planning B-Planning & Design,* Vol:37, pp.982-1001.

Psarra, S (2010) 'The Ghost of Conceived Space – What Kind of Work Does or Should Space Syntax Perform for Architecture', *The Journal of Space Syntax*, 1(1) pp. 17-29.

Schaffranek, R., Vasku, M. (2013) 'Space Syntax for Generative Design: On the Application of a New Tool', In: *Proceedings of the 9th International Space Syntax Symposium*, Seoul, pp.50.1-50.12.

Schön, A. D. (1987) *Educating the Reflective Practitioner*. San Francisco: Jossey-Bass Publishers.

Space Syntax Laboratory (1997) 'A Brief Lesson in Pesh', *Lecture Notes, MSc Advanced Architectural Studies,* UCL Bartlett School of Graduate Studies, 2004-2005.

Space Syntax Laboratory (1997) 'An Advance Tutorial in Axman', *Lecture Notes, MSc Advanced Architectural Studies,* UCL Bartlett School of Graduate Studies, 2004-2005.

Sevtsuk, A., Mekonnen, M. (2012) 'Urban Network Analysis: A New Toolbox for Measuring City Form in ArcGIS' *Proceedings of the 2012 Symposium on Simulation for Architecture and Urban Design,* Orlando, USA, 26-30 March 2012, pp.1-10.

Turner, A. (eds.) (2007) 'New developments in Space Syntax Software', In: 6*th International Space Syntax Symposium*, Istanbul Technical University: Istanbul, Turkey.

Turner, A., Penn, A., Hillier, B. (2005) 'An algorithmic Definition of the Axial Map', *Environment and Planning B: Planning and Design*, 32(3), pp.425-444.

Wineman J., Turner J., Psarra S., Senske N., Jung S. (2007) 'Syntax 2D: An Open-Source Software Platform for Space Syntax Analysis', In: *New Developments in Space Syntax Software, ed. A. Turner, 6th International Space Syntax Symposium,* Istanbul, Turkey, 12-15 June 2007, pp. 23-26.

Varoudis, T., Psarra, S. (2014) 'Beyond Two Dimensions: Architecture Through Three Dimensional Visibility Graph Analysis', *The Journal of Space Syntax*, Vol 5. pp.91-108.

Yongzhi L., Lertlakkhanakul J., Lee, S., Choi, J. (2009) 'Design with Space Syntax Analysis Based on Building Information Model: Towards an Interactive Application of Building Information Model in Early Design Process', *Joining Languages, Cultures, and Visions: CAAD Futures,* T. Tidafi and T. Dorta (eds.), Montreal, pp.501-514.

URL-1: https://www.spacesyntax.net/software/

URL-2: https://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=spacesyntax;9b5f08f1.00

URL-3: http://www.complexcity.info/media/software/