Consistency Checker

An automatic constraint-based evaluator for housing spatial configurations

Hardik Arora[1], Jessica Bielski[2], Viktor Eisenstadt[3], Christoph Langenhan[4], Christoph Ziegler[5], Klaus-Dieter Althoff[6], Andreas Dengel[7]
[1]Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI) / Go Reply GmbH [2,4,5]Technical University of Munich (TUM) [3,6]Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI) / University of Hildesheim [7]Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI)
[1]h.arora@reply.de [2,4,5]{jessica.bielski|langenhan|c.ziegler}@tum.de [3,6,7]{viktor.eisenstadt|klaus-dieter.althoff|andreas.dengel}@dfki.de

The gradual rise of artificial intelligence (AI) and its increasing visibility among many research disciplines affected Computer-Aided Architectural Design (CAAD). Architectural deep learning (DL) approaches are being developed and published on a regular basis, such as retrieval (Sharma et al. 2017) or design style manipulation (Newton 2019; Silvestre et al. 2016). However, there seems to be no method to evaluate highly constrained spatial configurations for specific architectural domains (such as housing or office buildings) based on basic architectural principles and everyday practices. This paper introduces an automatic constraint-based consistency checker to evaluate the coherency of semantic spatial configurations of housing construction using a small set of design principles to evaluate our DL approaches. The consistency checker informs about the overall performance of a spatial configuration followed by whether it is open/closed and the constraints it didn't satisfy. This paper deals with the relation of spaces processed as mathematically formalized graphs contrary to existing model checking software like Solibri.

Keywords: *model checking, building information modeling, deep learning, data quality*

INTRODUCTION

The development and design approaches of architecture are characterized by continuous change due to social, ecological, and furthermore technological conditions. Contemporary and future design quality assessment take place against the background of these ever-changing framework conditions. Inspired by system theory, the first computer-based approaches were developed revolving around design rules (Purcell et al. 1990), but were unable to exhaustively formalize the complexity of architectural designs. The second generation of the design methodology movement in the 1970s, represented by Horst Rittel and others, viewed the design process as an individual process that could only be described incompletely (Richter 2010). From the current perspective, the formalization of complex cases in architecture is not sufficiently solved and is referred to as data acquisition bottleneck. To remedy these shortcomings, Langenhan (2010;2013;2017) introduced the novel approach 'semantic building fingerprints' that facilitates spatial relationships. The digital semantic fingerprint of buildings describes the main semantic features of design, forming the basis for similarity assessment to deal with ambiguities and complexities of architecture. However, beginning in the 1980s, the digital approaches of case-based reasoning (CBR) have been introduced to the field of Computer-Aided Architectural Design (CAAD), influencing the building design and leading to artificial intelligence (AI) research in the building industry in the form of case-based design as early as the 1990s. Because of the gradual rise of AI and its increasing visibility architectural deep learning (DL) approaches are being developed and published on a regular basis such as retrieval (Sharma et al. 2017) or design style manipulation (Newton 2019; Silvestre et al. 2016). The original architectural design process without any computational support suggests a wide range of different designs as inspiration (Richter 2010) and for generating alternatives as a common practice, which can consequently be identified as an incremental learning process with iterative evaluation by the architect (Elango and Devadas 2014a). Inspired by the works of Elango and Devadas (2014a), a DL approach is developed to suggest design variations, based on a large dataset of reference buildings, for architects in the early design phase. However, there seems to be no method to evaluate highly constrained spatial configurations for specific architectural domains based on basic architectural principles and everyday practices. The necessary data quality for a DL approach not only refers to quantitative pieces of information but also refers to the qualitative aspects. The lack of such an evaluator can have severe effects on the qualitative performance of different DL approaches. This paper introduces an automatic constraint-based consistency checker to evaluate the coherency of spatial configurations of housing construction according to a set of rules based on general principles of architecture as the data used for training the DL approach should be coherent with architectural principles and everyday practices.

PROBLEM STATEMENT

The process of designing spatial configurations is an amalgamation of ill-defined design problems and non-linear decision-making. Horst Rittel and Melvin Weber (1973) stated that design problems themselves are wicked problems as they can't be definitively described. "The design process is complex due to its content, context, stakeholders, ill-defined problems, and multifaceted interactions. Furthermore, each design process has special characteristics which are not easily standardized. Gann et al. (2003) stated the difficulty of quantifying the quality of architectural design since it consists of both tangible and intangible facts and objective-subjective components." (Harputlugil et al. 2014, p.139). Simultaneously, the decision-making processes in architecture are non-linear and highly interactive. It is a mix of rational and intuitive decisions, and there are no step-by-step procedures (Elango and Devadas 2014b, p.1). Architectural data is complex and error-prone due to the difficulties in defining both design problems and design decision-making, as mentioned above. It is necessary to ensure the data quality if it is used for training Machine Learning (ML) or Deep Learning (DL) approaches. The quality of data determines the performance of any ML or DL approach (Sessions and Valtorta 2006). The data quality not only refers to quantitative pieces of information but also refers to the qualitative aspects. The lack of such an evaluator can have severe effects on the qualitative performance of different DL approaches.

RESEARCH CONTEXT & RELATED WORK

During the currently running research project [metis-II] (2020-2023), we examine and develop DL-based methods and approaches for the support of the early conceptual phases in architectural design. Taking into account the vagueness and uncertainty of architectural design data in the form of graph-based spatial configurations, we investigate how auto-completion of floor plans (comparable to, for example, sentence completion on modern mobile gadgets) can be achieved using artificial neural networks. Based on early sketches of the building designs, rooms and the possible relations between them are suggested to the architect to enhance the early ideation process. The auto-completion methods are intended to be a helpful tool for architects during the early conceptual process to help them overcome design bias. Providing them with the different design continuation options is intended to create interaction patterns to assess their own design decisions and examine the possible further development of the current spatial configuration state. Successful approaches for purposes other than ours use architectural image data: as examples, search for similar designs (Sharma et al. 2017), modification of the design style (Newton 2019; Silvestre et al. 2016), or estimation of the layout in 3D (Sun et al. 2019) can be named. Even though new DL approaches are researched in the field of architecture, there still lacks a method to evaluate the architectural design quality to ensure good data quality for training different DL approaches.

Architectural design quality assessment is typically done through post-occupancy evaluation (POE), which is a systematic evaluation of the occupant after a certain period of time of inhabitation. However, created for satisfaction as an empirical basis to improve existing buildings, it is deemed unfit for early design stages (Harputlugil et al. 2014). Different strategies, such as the Design Quality Indicator (DQI) (CIC 1999) and Laseau's (2000) variables for typical design problems, have been applied to assess the design quality of architectural design decisions in the early phases. The design decisions and prioritizing of the different variables within these first design phases have a significant impact on ensuring the quality of the end product. In the same way as the "information on design quality important is [during these early stages] to expanding the capabilities of the design team to make well-informed choices" (Harputlugil et al. 2014, p.140), it is needed to ensure the data quality for deep learning. Drawing from the Vitruvius' principles, the DQI, and Laseau's (2000) variables, Harputlugil et al. (2014) divide and criterias of the architectural design process as follows: functionality, built quality, and impact, depending on the culture, society, and technology of the era. The functionality has been shown to be further divided and prioritized by use e.g., fit for functionality, access e.g., local and interior access, and finally space e.g., relation of spaces. Overall, the architectural design decision process can be depicted as an Analytic Hierarchy Process (AHP) with Multi-Criteria Decision Making (MCDM), focusing on the use of the building, its individual spaces, and their interrelations, in the early design phases.

Constraint-based approaches are generally used in architecture to design spatial configurations. Layout planning is a historical design activity that affects the characteristics and performance of a building throughout its lifecycle (Shikdar et al. 2010). According to the authors, the design constraints are a guide to search spatial solutions. Constraint-based approaches allow the designer to interact with the layout planning process, which simulates the iterative nature of the creative design (Shikdar et al. 2010).

Damski and Gero (2006) suggested a system to develop space layout topologies for spatial configurations using an evolutionary approach. They considered spatial layouts as a set of topological and directional constraints, which was used as a fitness function in the evolutionary system.

The constraint-based approaches mentioned above only provide a solution to the design problems of spatial topologies. However, currently, no constraint-based approaches are used to evaluate a semantic spatial configuration. In this work, a constraint-based consistency checker was developed to ensure coherency of spatial configurations (see figure 1) based on quantifiable and countable criteria, which results in improved data quality and quantity for training different DL approaches.

APPROACH

This paper introduces an automatic constraint-based consistency checker to evaluate the coherency of semantic spatial configurations of housing construction according to a set of rules. The consistency checker informs about the overall performance of a semantic spatial configuration followed by whether it is an open floor plan ('open'/'closed') and the constraints it didn't satisfy. Semantic spatial configuration informs about the semantics associated with a spatial configuration (i.e., building floor plan). Semantics refers to the information regarding different connection types connecting different room types in a spatial configuration.In our current work on a DL-based spatial layout auto-completion system that makes recommendations for architects in the early design phases, data quality plays an essential role.

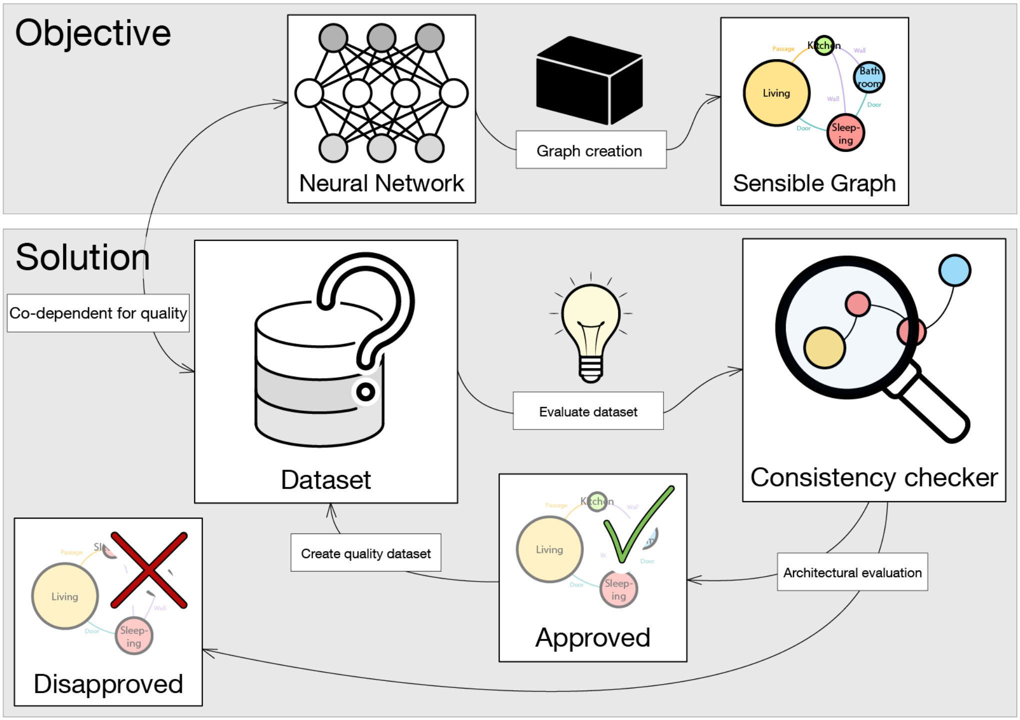


Figure 1
Main objective and approach to ensure data quality.

The data used for training the DL approach should be coherent, otherwise, the auto-completion system can recommend rooms, and connections that seem implausible to architects, e.g., connect a bedroom with a kitchen via a window. Due to the lack of such an evaluator, the consistency checker was developed. The semantic spatial configurations are formalized as graphs, where the nodes of the graph represent the room types, and the edges between the nodes of the graph represent the connection types. We developed a set of 11 rules to evaluate the consistency of the data at our disposal. The main idea behind these 11 rules is to omit the semantic spatial configurations which don't adhere to the general rules from the final dataset (dataset for training DL approaches). Each spatial configuration has to pass each of the consistency rules. The consistency rules are the following:

- No spatial configuration should contain a room with no connection to any other room.
- No spatial configuration should contain the room-type building services.
- Every spatial configuration must contain at least one bathroom/toilet, one kitchen, and one sleeping/living/generic room, except if it is a one-room-apartment.
- Every spatial configuration must contain at least three rooms.
- A passage shouldn't be allowed to connect a kitchen and a sleeping room.
- A passage from the bathroom/toilet should only be allowed when it connects to a sleeping room.
- No spatial configuration should have direct access (i.e., via passage/door) between the bathroom/toilet and kitchen.
- Each room should have at least an edge connection to another room using a door, passage, entrance, or wall.
- If a room connects with another room via a wall, it must connect to another room via a door/passage.
- There should be no passage connection between the living room and bathroom/toilet via another room, e.g., bathroom-passage- sleeping room-passage-living room.
- If a room has only one connection type, it should be a passage, door, or entrance. If a room has more than one connection, it needs to have at least two direct connections, i.e., via passage/-door, except if the connection type is the entrance.

A web-based tool was developed to ensure that architects can reexamine the rules suggested for the consistency checker. The web-based tool displays information regarding the spatial configurations present in the database in a tabular format. Each counted row starts with the name of the Architectural GraphML (AGraphML) file (Langenhan, 2017). This file is taken from a dataset that was previously created by combining different databases for e.g. geometric, topological and lexical data (Roith, Langenhan & Petzold, 2019). The following column shows the consistency score, i.e., the number of rules passed by the spatial configuration. Following, the number of checks the spatial configuration failed are mentioned within square brackets. The next column presents the number of rooms in the spatial configuration, and finally whether a spatial configuration is open/closed (see figure 2). Furthermore, the graph-based representation (i.e., image) of the spatial configuration is depicted, which is created using a Python module known as Graph-tool. The last four columns of the web-based tool consist of radio buttons for each architect - in this scenario: the two architects A1 and A2 - to approve or disapprove. Each architect evaluates two different topics within their respective column:

- Does the spatial configuration of the graph visualisation adhere to architectural principles for floor plan layouting of residential housing?
- Is the apartment type, consisting of the room count for habitable rooms and floor type ('open'/'closed'), correctly labelled by the system?

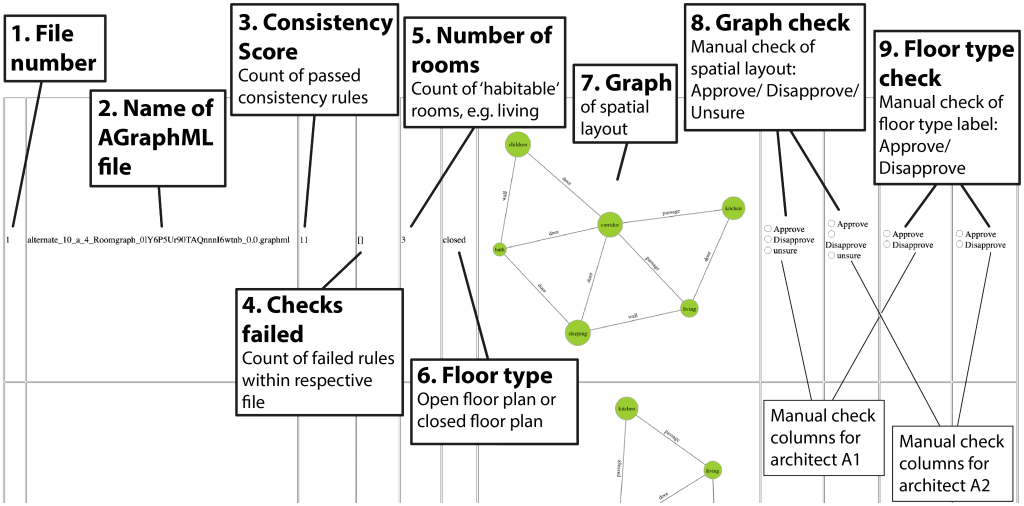


Figure 2
Overview of the columns of the web-based tool.

Architects also have an option to select 'Unsure' for evaluating the spatial configuration about whether it should be included or not in the final dataset. Therefore, they could separately discuss the said floor plan.

If none of the architects select an option (i.e., approval or disapproval of a spatial configuration for the inclusion in the final dataset), then the row representing the spatial configuration has a white background. If both the architects approve it, then the row's background color should be green (see figure 3). If both the architects disapprove of it, then the row's background color should be red (see figure 3).

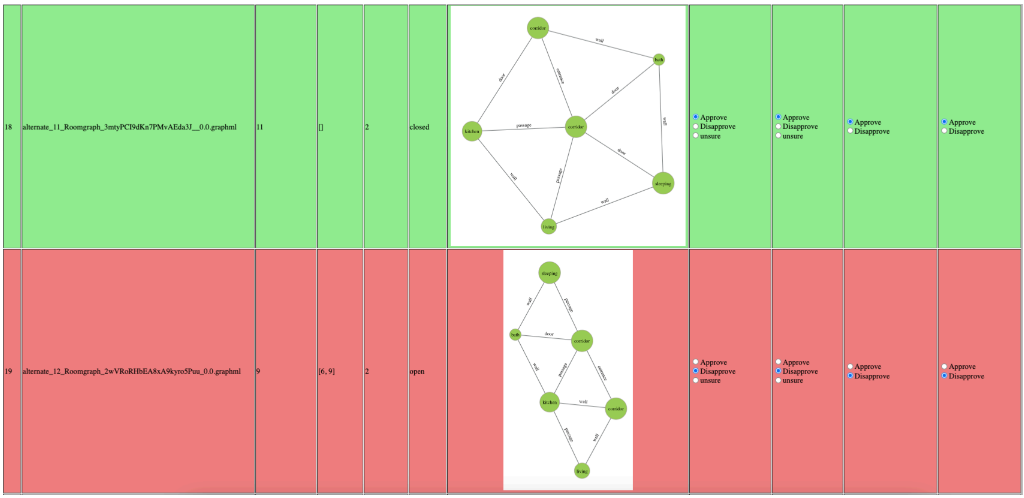
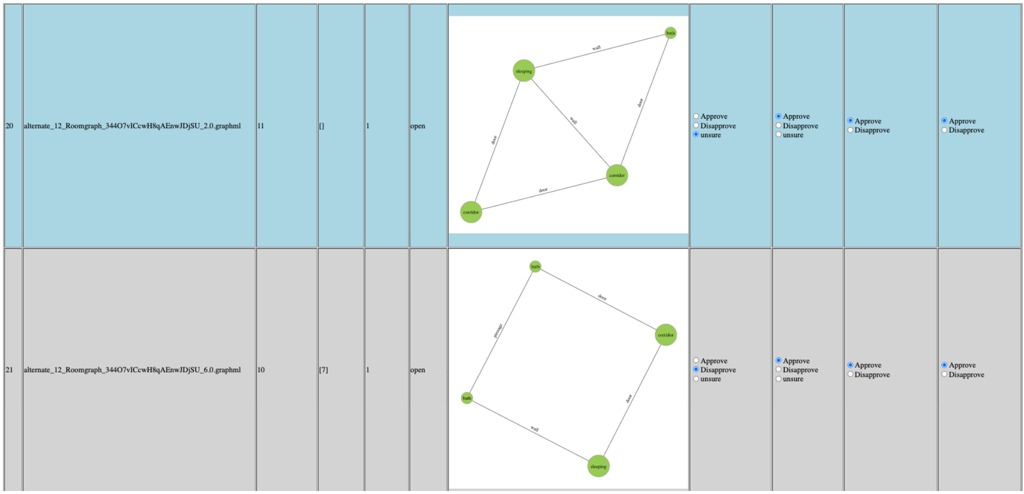


Figure 3
The web-based tool with both architects either approving or disapproving the spatial configuration.

Figure 4
The web-based tool with either of the architects is unsure or there is a disagreement regarding the approval or disapproval of the spatial configuration.



If both the architects have different opinions, i.e., one approves, and the other disapproves, then the row's background color should be grey (see figure 4). If either of the architects is unsure of it, then the row's background color should be blue (see figure 4), and the name of the AgraphML gets added to a different list accessible via "List of unsure files" on the homepage.

As mentioned above, the consistency checker evaluated semantic spatial configurations, followed by a manual evaluation by architects of our research group. The dual evaluation ensured that spatial configurations approved by both the architects and the consistency checker are in the final dataset. After the first round of manual evaluation of spatial configurations the rules of the consistency checker were reexamined. Due to the reexamination, we were able to revise the consistency rules for the second iteration. The second iteration with additional and improved consistency rules resulted in more approved spatial configurations for the final dataset.

A total of 597 AgraphML files representing the semantic spatial configurations were evaluated using the consistency checker with the revised set of rules. The consistency checker approved 286 AgraphML files, which were also approved by the architects of our research group.

CONCLUSION AND FUTURE WORK

The main idea of this work is to ensure the use of semantic spatial configurations for training different DL approaches and check the results. The consistency checker is an important module for the application of DL in architecture. Since a trained neural network is a black box, the consistency checker helps in evaluating the results of the trained neural network. In this work, we presented an automatic constraint-based consistency checker to evaluate the coherency of semantic spatial configurations of housing construction according to a set of rules. The data evaluation of the consistency checker and manual evaluation performed by the architects was found to be coherent. However, the rules of the consistency checker were formalized by the architects of our research group, who also evaluated the consistency checker.

Additionally, the proposed constraint-based consistency checker will be integrated into an existing DL pipeline as a use case, which allows both the training data and the recommendations of the

Neural Network to be validated by the consistency checker. In our current work on a DL-based spatial layout autocompletion system (metis-II) that makes recommendations for architects in the early design phases, data quality plays an essential role. The maximum number of architects using the web-based tool is currently two , while in the future, we aim to scale the web-tool to accommodate a larger diaspora. The updated version of the web-based tool will also allow architects to add new constraints.

REFERENCES

Damski, JC and Gero, JJ 1997 'AN EVOLUTIONARY APPROACH TO GENERATING CONSTRAINT-BASED SPACE LAYOUT TOPOLOGIES', *CAAD Futures 1997*

Elango, M and Devadas, M 2014 'Identification of Parameters for Evaluating Architectural Design. - A Case of Housing in Hot and Humid Region, Chennai City, India', *Proceedings of ARCHDESIGN / Architectural Design Conference on Design Methodologies*

Elango, M and Devdas, M 2014, 'Multi-Criteria Analysis of the Design Decisions In Architectural Design Process during the Pre-Design Stage', *International Journal of Engineering and Technology*, 6, pp. 1033-1046

Gann, D, Salter, A and Whyte, J 2003, 'Design Quality Indicator as a tool for thinking', *Building Research and Information*, 31, pp. 318-333

Harputlugil, T, Gultekin, A, Prins, M and Topcu, YI 2014, 'Architectural Design Quality Assessment Based On Analytic Hierarchy Process: A Case Study', *Journal of The Faculty of Architecture*, 3, pp. 139-161

Langenhan, C 2017, *Datenmanagement in der Architektur. Untersuchung zur Organisation von Entwurfsinformationen in IT-Infastrukturen und Nutzungsmöglichkeiten in wissensbasierten Systemen*, Ph.D. Thesis, Technische Universität München

Langenhan, C and Petzold, F 2010, 'The Fingerprint of Architecture: Sketch-Based Design Methods of Researching Building Layouts through the Semantic Fingerprint Of FloorPlans', *Journal of Architecture and Modern Information Technologies*, 4, pp. 4-13

Langenhan, C and Petzold, F 2015 'BEYOND THE BUBBLE- Computer-aided topological analysis and parametric design of room configurations in university education', *Proceedings of eCAADe 2015*, Vienna

Langenhan, C, Weber, M, Liwicki, M, Petzold, F and Dengel, A 2013, 'Graph-based retrieval of building information models for supporting the early design stages', *Advanced Engineering Informatics*, 27(4), pp. 413-426

Laseau, P 2000, *Graphic Thinking for Architects and Designers*, John Wiley & Sons

Newton, D 2019, 'Generative Deep Learning in Architectural Design', *Technology|Architecture + Design*, 3, pp. 176-189

Purcell, PA, Mitchell, WJ and McCullough, M 1990, *The Electronic design studio: Knowledge and media in the computer era*, MIT Press

Richter, K 2010, *Augmenting Designers' Memory: Case Based Reasoning in der Architektur*, Ph.D. Thesis, Bauhaus-Universität Weimar

Rittel, HWJ and Weber, MM 1973, 'Dilemmas in a General Theory of Planning', *Policy Sciences*, 4, pp. 155-169

Roith, J, Langenhan, C and Petzold, F 2017, 'Supporting the building design process with graph-based methods using centrally coordinated federated databases', *Visualization in Engineering*, 5, p. Article 20

Sessions, V and Valtorta, M 2006 'The Effects of Data Quality on Machine Learning Algorithms', *Proceedings of ICIQ 2006*

Sharma, D, Gupta, N, Chattopadhyay, C and Mehta, S 2017, 'DANIEL: A Deep Architecture for Automatic Analysis and Retrieval of Building Floor Plans', *14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, 01, pp. 420-425

Shikder, S, Price, A and Mourshed, M 2010 'Interactive Constraint-Based Space Layout Planning', *no source given*

Silvestre, J, Ikeda, F and Guéna, F 2016 'Artificial imagination of architecture with deep convolutional neural network "Laissez-faire": Loss of control in the esquisse phase', *CAADRIA 2016, 21st International Conference on Computer-Aided Architectural Design Research in Asia - Living Systems and Micro-Utopias: Towards Continuous Designing*

Sun, C, Hsiao, CW, Sun, M and Chen, HT 2019 'Horizonnet: Learning room layout with 1d representation and pano stretch data augmentation', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2019*

[1] https://www.dqi.org.uk/