The aim of these workshops and conference is to help transfer and spread newly appearing design technologies, educational methods and digital modelling supported by information technology in architecture. By organizing a workshop with a conference, we would like to close the distance between practice and theory.

Architects who keep up with the new designs demanded by the building industry will remain at the forefront of the design process in our information-technology based world. Being familiar with the tools available for simulations and early phase models will enable architects to lead the process. We can get "back to command".

The other message of our slogan is "Back to command". In the expanding world of IT applications there is a need for the ready change of preliminary models by using parameters and scripts. These approaches retrieve the feeling of command-oriented systems.
Editor

Mihály Szoboszlai
Faculty of Architecture
Budapest University of Technology and Economics

2nd edition, July 2016

CAADence in Architecture – Proceedings of the International Conference on Computer Aided Architectural Design, Budapest, Hungary, 16th-17th June 2016. Edited by Mihály Szoboszlai, Department of Architectural Representation, Faculty of Architecture, Budapest University of Technology and Economics

Cover page: Faraway Design Kft.

Layout, typography: based on proceedings series of eCAADe conferences

DTP: Tamás Rumi

ISBN: 978-963-313-225-8
ISBN: 978-963-313-237-1 (online version)

CAADence in Architecture. Back to command
Budapesti Műszaki és Gazdaságtudományi Egyetem

Copyright © 2016

Publisher: Faculty of Architecture, Budapest University of Technology and Economics

All rights reserved. No part of this publication may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of the publisher.
CAADence in Architecture
Back to command

Proceedings of the International Conference on Computer Aided Architectural Design

16-17 June 2016
Budapest, Hungary
Faculty of Architecture
Budapest University of Technology and Economics

Edited by
Mihály Szoboszlai
Theme

CAADence in Architecture
Back to command

The aim of these workshops and conference is to help transfer and spread newly appearing design technologies, educational methods and digital modelling supported by information technology in architecture. By organizing a workshop with a conference, we would like to close the distance between practice and theory. Architects who keep up with the new design demanded by the building industry will remain at the forefront of the design process in our IT-based world. Being familiar with the tools available for simulations and early phase models will enable architects to lead the process. We can get “back to command”.

Our slogan “Back to Command” contains another message. In the expanding world of IT applications, one must be able to change preliminary models readily by using different parameters and scripts. These approaches bring back the feeling of command-oriented systems, although with much greater effectiveness.

Why CAADence in architecture?

“The cadence is perhaps one of the most unusual elements of classical music, an indispensable addition to an orchestra-accompanied concerto that, though ubiquitous, can take a wide variety of forms. By definition, a cadence is a solo that precedes a closing formula, in which the soloist plays a series of personally selected or invented musical phrases, interspersed with previously played themes – in short, a free ground for virtuosic improvisation.”

Nowadays sophisticated CAAD (Computer Aided Architectural Design) applications might operate in the hand of architects like instruments in the hand of musicians. We have used the word association cadence/caadence as a sort of word play to make this event even more memorable.

Mihály Szoboszlai
Chair of the Organizing Committee
Sponsors

GRAPHISOFT
ARCHICAD

AUTODESK

STUDIO IN-EX
ARCHITECTS & ENGINEERS

MŰEGYETEM 1782

Építészeti Ábrázolás Tanszék
Department of Architectural Representation
Acknowledgement

We would like to express our sincere thanks to all of the authors, reviewers, session chairs, and plenary speakers. We also wish say thank you to the workshop organizers, who brought practice to theory closer together.

This conference was supported by our sponsors: GRAPHISOFT, AUTODESK, and STUDIO IN-EX. Additionally, the Faculty of Architecture at Budapest University of Technology and Economics provided support through its “Future Fund” (Jövő Alap), helping to bring internationally recognized speakers to this conference.

Members of our local organizing team have supported this event with their special contribution – namely, their hard work in preparing and managing this conference.

Mihály Szoboszlai
Chair of the Organizing Committee

Local conference staff
Ádám Tamás Kovács, Bodó Bánáti, Imre Batta, Bálint Csabay, Benedek Gáspor, Alexandra Gőőz, Péter Kaknics, András Zsolt Kovács, Erzsébet Kőnigné Tóth, Bence Krajnyák, Levente Lajtos, Pál Ledneczki, Mark Searle, Béla Marsal, Albert Máté, Boldizsár Medvey, Johanna Pék, Gábor Rátonyi, László Strommer, Zsanett Takács, Péter Zsigmond
Workshop tutors

Algorithmic Design through BIM
   Erik Havadi
   Laura Baróthy

Working with BIM Analyses
   Balázs Molnár
   Máté Csócsics
   Zsolt Oláh

OPEN BIM
   Ákos Rechtorisz
   Tamás Erős

GDL in Daily Work
   Gergely Fehér
   Dominika Bobály
   Gergely Hári
   James Badcock
List of Reviewers

Abdelmohsen, Sherif - Egypt
Achten, Henri - Czech Republic
Agkathidis, Asterios - United Kingdom
Asanowicz, Aleksander - Poland
Bhatt, Anand - India
Braumann, Johannes - Austria
Celani, Gabriela - Brazil
Cerovsek, Tomo - Slovenia
Chaszar, Andre - Netherlands
Chronis, Angelos - Spain
Dokonal, Wolfgang - Austria
Estévez, Alberto T. - Spain
Fricker, Pia - Switzerland
Herr, Christiane M. - China
Hoffmann, Miklós - Hungary
Juhász, Imre - Hungary
Jutraz, Anja - Slovenia
Kieferle, Joachim B. - Germany
Klinc, Robert - Slovenia
Koch, Volker - Germany
Kolarevic, Branko - Canada
König, Reinhard - Switzerland
Krakhofer, Stefan - Hong Kong
van Leeuwen, Jos - Netherlands
Lomker, Thorsten - United Arab Emirates
Lorenz, Wolfgang - Austria
Loveridge, Russell - Switzerland
Mark, Earl - United States
Molnár, Emil - Hungary
Mueller, Volker - United States
Nourian, Pirouz - Netherlands
Oxman, Rivka - Israel
Parlac, Vera - Canada
Quintus, Alex - United Arab Emirates
Searle, Mark - Hungary
Szoboszlai, Mihály - Hungary
Tuncer, Bige - Singapore
Verbeke, Johan - Belgium
Vermillion, Joshua - United States
Watanabe, Shun - Japan
Wojtowicz, Jerzy - Poland
Wurzer, Gabriel - Austria
Yamu, Claudia - Netherlands
Contents

14 Keynote speakers

15 Keynote
15 Backcasting and a New Way of Command in Computational Design
Reinhard Koenig, Gerhard Schmitt

27 Half Cadence: Towards Integrative Design
Branko Kolarevic

33 Call from the industry leaders
33 Kajima’s BIM Theory & Methods
Kazumi Yajima

41 Section A1 - Shape grammar
41 Minka, Machiya, and Gassho-Zukuri
Procedural Generation of Japanese Traditional Houses
Shun Watanabe
49 3D Shape Grammar of Polyhedral Spires
László Strommer

55 Section A2 - Smart cities
55 Enhancing Housing Flexibility Through Collaboration
Sabine Ritter De Paris, Carlos Nuno Lacerda Lopes
61 Connecting Online-Configurators (Including 3D Representations) with CAD-Systems
Small Scale Solutions for SMEs in the Design-Product and Building Sector
Matthias Kulcke
67 BIM to GIS and GIS to BIM
Szabolcs Kari, László Lellei, Attila Gyulai, András Sik, Miklós Márton Riedel
<table>
<thead>
<tr>
<th>Page</th>
<th>Section</th>
<th>Title</th>
<th>Authors</th>
</tr>
</thead>
<tbody>
<tr>
<td>73</td>
<td>A3</td>
<td>Modeling with scripting</td>
<td>Bálint Péter Füzes, Dezső Hegyi</td>
</tr>
<tr>
<td>79</td>
<td></td>
<td>De-Script-ion: Individuality / Uniformity</td>
<td>Helen Lam Wai-yin, Vito Bertin</td>
</tr>
<tr>
<td>87</td>
<td>B1</td>
<td>BIM</td>
<td>Michio Matsubayashi, Shun Watanabe</td>
</tr>
<tr>
<td>93</td>
<td></td>
<td>Integration of Facility Management System and Building Information Modeling</td>
<td>Lei Xu</td>
</tr>
<tr>
<td>99</td>
<td></td>
<td>BIM as a Transformer of Processes</td>
<td>Ingolf Sundfør, Harald Selvær</td>
</tr>
<tr>
<td>105</td>
<td>B2</td>
<td>Smooth transition</td>
<td>Szilvia B.-S. Béla, Márta Szilvási-Nagy</td>
</tr>
<tr>
<td>111</td>
<td></td>
<td>A General Theory for Finding the Lightest Manmade Structures Using Voronoi and Delaunay</td>
<td>Mohammed Mustafa Ezzat</td>
</tr>
<tr>
<td>119</td>
<td>B3</td>
<td>Media supported teaching</td>
<td>Pia Fricker</td>
</tr>
<tr>
<td>127</td>
<td></td>
<td>The Importance of Connectivism in Architectural Design Learning: Developing Creative Thinking</td>
<td>Verónica Paola Rossado Espinoza</td>
</tr>
<tr>
<td>133</td>
<td></td>
<td>Ambient PET(b)ar</td>
<td>Kateřina Nováková</td>
</tr>
<tr>
<td>141</td>
<td></td>
<td>Geometric Modelling and Reconstruction of Surfaces</td>
<td>Lidija Pletenac</td>
</tr>
</tbody>
</table>
149 Section C1 - Collaborative design + Simulation

155 Horizontal Load Resistance of Ruined Walls Case Study of a Hungarian Castle with the Aid of Laser Scanning Technology
Tamás Ther, István Sajtos

163 2D-Hygrothermal Simulation of Historical Solid Walls
Michela Pascucci, Elena Lucchi

169 Responsive Interaction in Dynamic Envelopes with Mesh Tessellation
Sambit Datta, Smolik Andrei, Tengwen Chang

169 Identification of Required Processes and Data for Facilitating the Assessment of Resources Management Efficiency During Buildings Life Cycle
Moamen M. Seddik, Rabee M. Reffat, Shawkat L. Elkady

177 Section C2 - Generative Design - 1

177 Stereotomic Models In Architecture A Generative Design Method to Integrate Spatial and Structural Parameters Through the Application of Subtractive Operations
Juan José Castellón González, Pierluigi D’Acunto

185 Visual Structuring for Generative Design Search Spaces
Günsu Merin Abbas, İpek Gürsel Dino

195 Section D2 - Generative Design - 2

195 Solar Envelope Optimization Method for Complex Urban Environments
Francesco De Luca

203 Time-based Matter: Suggesting New Formal Variables for Space Design
Delia Dumitrescu

213 Performance-oriented Design Assisted by a Parametric Toolkit - Case study
Bálint Botzheim, Kitti Gidófalvy, Patricia Emy Kikunaga, András Szollár, András Reith

221 Classification of Parametric Design Techniques
Types of Surface Patterns
Réka Sárközi, Péter Iványi, Attila Béla Széll
Section D1 - Visualization and communication

Issues of Control and Command in Digital Design and Architectural Computation
Andre Chaszar

Integrating Point Clouds to Support Architectural Visualization and Communication
Dóra Surina, Gábor Bödő, Konsztantinosz Hadzijanisz, Réka Lovas, Beatrix Szabó, Barnabás Vári, András Fehér

Towards the Measurement of Perceived Architectural Qualities
Benjamin Heinrich, Gabriel Wurzer

 Complexity across scales in the work of Le Corbusier
Using box-counting as a method for analysing facades
Wolfgang E. Lorenz

Author’s index
Keynote speakers

REINHARD KÖNIG
Reinhard König studied architecture and urban planning. He completed his PhD thesis in 2009 at the University of Karlsruhe. Dr. König has worked as a research assistant and appointed Interim Professor of the Chair for Computer Science in Architecture at Bauhaus-University Weimar. He heads research projects on the complexity of urban systems and societies, the understanding of cities by means of agent based models and cellular automata as well as the development of evolutionary design methods. From 2013 Reinhard König works at the Chair of Information Architecture, ETH Zurich. In 2014 Dr. König was guest professor at the Technical University Munich. His current research interests are applicability of multi-criteria optimisation techniques for design problems and the development of computational analysis methods for spatial configurations. Results from these research activities are transferred into planning software of the company DecodingSpaces. From 2015 Dr. König heads the Junior-Professorship for Computational Architecture at Bauhaus-University Weimar, and acts as Co-PI at the Future Cities Lab in Singapore, where he focus on Cognitive Design Computing.
Main research project: Planning Synthesis & Computational Planning Group see also the project description: Computational Planning Synthesis and his external research web site: Computational Planning Science

BRANKO KOLAREVIC
Branko Kolarevic is a Professor of Architecture at the University of Calgary Faculty of Environmental Design, where he also holds the Chair in Integrated Design and co-directs the Laboratory for Integrative Design (LID). He has taught architecture at several universities in North America and Asia and has lectured worldwide on the use of digital technologies in design and production. He has authored, edited or co-edited several books, including “Building Dynamics: Exploring Architecture of Change” (with Vera Parlac), “Manufacturing Material Effects” (with Kevin Klinger), “Performative Architecture” (with Ali Malkawi) and “Architecture in the Digital Age.” He is a past president of the Association for Computer Aided Design in Architecture (ACADIA), past president of the Canadian Architectural Certification Board (CACB), and was recently elected future president of the Association of Collegiate Schools of Architecture (ACSA). He is a recipient of the ACADIA Award for Innovative Research in 2007 and ACADIA Society Award of Excellence in 2015. He holds doctoral and master’s degrees in design from Harvard University and a diploma engineer in architecture degree from the University of Belgrade.
Visual Structuring for Generative Design Search Spaces

Günsu Merin Abbas¹, İpek Gürsel Dino²
¹The Department of Architecture
TOBB University of Economy and Technology, Turkey
e-mail: gmabbas@etu.edu.tr
²The Department of Architecture
Middle East Technical University, Turkey
e-mail: ipekg@metu.edu.tr

Abstract: With generative design strategies, the act of design, the problem-solving process, and the interaction, understanding and representation of design artefacts have changed. As such strategies entail a large number of possible design solutions, they also expand the design search space immensely. In addition, due to the automated design generation process, the interaction of the designer through the design process has decreased. This research problematizes two major points as (1) the broad design spaces of generative design systems that are populated by many design instances and (2) the elimination of designerly decisions and evaluation from the design artefact due to automated generation processes. As a solution to these problems, a visual structuring strategy is proposed. This strategy aims to act as a mediator between the designer and the design artefact during the generative process and manage the complexity resulting from the multiplicity of solutions. For visual structuring, three methods are proposed and classified as (1) perception-based visual structuring, (2) retrieval-based structuring, and (3) optimality-based structuring within the scope of this research.

Keywords: generative design systems, design search spaces, visual structures, genetic algorithms, design variations

DOI: 10.3311/CAADence.1630

INTRODUCTION AND THE BASIC CONCEPTS

This paper discusses strategies developed to manage the complexity of large solution spaces of generative design systems (GDS) through visual structuring. In computational design synthesis, the understanding of a design process differs from the conventional design thinking. The act of design, the designer’s role in the design, as well as the notion of a design problem and the design artefact have altered. Before anything else, in computational design, design problems –that are inherently ill-structured– have to be reformulated into clearly defined problems. The design act is transformed into a continuous process of generation, transformation, and evaluation led by algorithms. In non-computational design exploration processes, the design solution space is formed manually by the designer. This limits the number of design alternatives, which narrows the design search spaces compared to the design search
spaces of GDS [1]. Here, the designer may need to explore and evaluate several design alternatives, which may prevent too-early design decisions. At this stage, to overcome such premature design decisions, GDS broadens the design search space by generating multiple design solutions at once. However, the high number of generated design solutions may challenge the management and organization in design search spaces.

With these changes, the designer’s interaction with the ‘artefact(s)’ has also changed. The designer’s contribution, evaluation and visual contact with the design process are weakened due to the expansion in design search space as a result of the automated design generation. Accordingly, this paper problematizes (1) the broad design search spaces of GDS that are populated by many design instances and (2) the elimination of designerly decisions from the generation phase due to automated generation, particularly of genetic algorithms (GA). We support that the design activity in generative design processes must be amplified to address subjective design exploration and visual evaluation. For this, there is a need for methods that facilitate the navigation of the designer through the solution space expanded by generative algorithms. As a solution, design representations such as visual structures are proposed as a mediator between the designer and design artefacts. Such representations have the potential to support the designerly evaluation of design instances and increase the designer’s involvement through the generation process. Three structuring strategies (perception-based, retrieval-based, and optimality-based) are discussed and compared.

Emergence

GDS demonstrate emergence as a consequence of complexity, one of non-linearity and self-organization [11,12]. Emergence is a change and reconfiguration of parts as a result of the underlying internal dynamics of a system that demonstrates itself through changes with recognizable and iterative patterns [11,13,14,15]. In complex systems, simple procedures determine the local interactions between the parts, resulting in emergent behaviour [15]. Those procedures are referred to as schema by Gell-Mann while Holland termed those procedures as internal model [29]. Within the scope of this paper, these procedures will be termed as schema. The schema is inherent to all design instances that are generated for a single design problem. As schema determine the internal structure of a system, the internal structure demonstrates itself with recognizable patterns as an emergent behaviour. In GA, within the framework of emergence, (1) genotype as the genetic composition of an organism associates with the idea of schema by designation of the generation logic while (2) phenotype, as the environmentally and genetically determined traits of an organism, associates with the emergent behaviour as an outcome of the process [16,17,18].

Generative Design Systems

Generative design systems are synthesis methods that facilitate design exploration, and entail divergence in the design search space by generating multiple design alternatives [2,3,4]. They simulate nature’s generation logic and foster the novel and efficient design exploration processes that extend the capabilities of designers and end up with un-repeatable design alternatives [5,6]. Within the scope of this paper, evolutionary systems, particularly GA, are investigated. Evolutionary systems simulate the processes of evolution in nature. In particular, GA are stochastic search mechanisms [7] that seek the fittest solution determined by the fitness function designating the fitness degree of the solution [8]. GA follows three operations; (1) representation (the cognitive modelling of a design problem, inputs), (2) generation (algorithms, operators, outputs), (3) evaluation and the guidance (the objective evaluation of generated design solutions, and the feedback) [4,9,10]. Through these steps, the designer becomes solely involved from the beginning of the representation phase by initializing the synthesis process; by defining a design problem, design constraints, design procedures and the boundary conditions [10]. This problem will be addressed by the strategies proposed in this paper in the following sections.
Visual Resemblance

In this paper, visual resemblance is accounted for by the common features of design instances that are generated from the same schema, and the common features are named as generic characteristics. The generic characteristics and resemblance associate with Arnheim’s “structural pattern” in visual reasoning and shape perception and also with Holland’s emergence with recognizable iterative patterns in GDS. Arnheim (1969) asserts that “[t]he perception of the shape is the grasping of structural features found in, or imposed upon, the stimulus material.” The perception of the shape is based on the general features that are perceived by the observer. Accordingly, there are two points to highlight: (1) perception is based on the main features that are captured and identified by the observer, and (2) such perception is observer-dependent, therefore, is subjective. Resemblance associates with the subjective recognition of a set of design instances. Hence, the design families are formed according to the designer’s evaluation and perception of generic characteristics through the population.

Resemblance, in relation to GDS, is a result of the schema that determines the main body of the population [11]. Each design instance is unique but resembles other design instances by sharing the same schema. In GA, as generic characteristics are determined using the genotype, new design variations emerge as the design variables change. Here, each instance is a variation of the schema; they are of the same type but are not identical. At this stage, this paper argues that design instances generated from the same schema may form a design family due to their similar phenotypic characteristics. Particularly in GA, the population is generated as an evolutionary lineage as they are of the same type [19].

STRUCTURING THE DESIGN SEARCH SPACE

For the main arguments, three structuring methods are discussed that may have the potential in (1) mediating between design generation and the subjectivity of the creative design by the amplification of the designer’s involvement in design synthesis via providing visual guidance and (2) to manage the complexity of large design search spaces.

Visual structuring is meant to give structure to the design search space that guides the interrelations between the design instances based on visual resemblance as a structuring criterion. Visual structures require the determination of the generic characteristics through a set of design instances. As forming categories is one of the major abilities of human cognition [20], the generic characteristics give way to the formation of resemblance groups, consequently, the classification of the design instances based on visual commonness. Visual structures can support design search and designer’s navigation through the space of possible solutions; this is particularly the case in GA, as the designer is in dialogue with the design artefact using symbolic representation, rather than visual. Visual structuring takes place after the generation process, which may assist the amplification of the designer’s interaction and the dialogue with design instances and the design search space. (Fig.1). Here, visual structuring has the potential to establish a visual dialogue between the designer and the process.

Different methods can support and improve the build-up of visual structures. Within the scope of this paper (1) perception-based, (2) retrieval-based, and (3) optimality-based structuring strategies are proposed and discussed.
Perception-Based Structuring

Perception-based structuring is an intuitive, case-based, observer-dependent and subjective act. It is based on the designer’s perception and identification of common features/visual resemblance between the design instances. Each structure is unique and personal. It requires manual build-up and is applicable for refined, manageable and small design search spaces. The designer identifies the hierarchical resemblance relationships between the design instances and groups them according to their visual similarity. Such groups are termed as resemblance clusters. All design instances can be represented in these visual structures, or filtration can be used to decrease the number of design instances for the formation of resemblance clusters. However, manual build-up may take a long time and require too much effort for a designer. Each parameter can lead to a separate cluster. In the case of multi-parameters, the act of classification is multi-phase; one cluster follows another, forming sub-clusters. Visually distinguishing the first parameter structures the whole classification, and this influences the whole taxonomy. As each designer has different visual perception, visually distinguishing parameters demonstrates differences for each designer. Therefore, each designer demonstrates different clustering behaviour and each time clustering behaviour changes according to the perception of a designer. (Fig.2).

To illustrate the formation of resemblance clusters, a small-scale case study was conducted. This case study exemplifies the classification behaviour/strategy of a designer rather than an implemented system. An existing design search space based on 110 unique chair designs, as experimented by Soddu, is used in which the design instances are generated by the evolutionary generative mechanism IDEA [5]. The solution set is given to three designers to form resemblance clusters and a brief statement for their grouping criteria. The designers used image-editing software to illustrate their clusters. Designer A selected the chair legs as a visually distinguishing feature of the design space, a criterion for the formation of resemblance clusters. Accordingly, she identifies eleven types of chair legs and the design search space is divided into eleven clusters. According to designer B, there are five resemblance clusters based on the form and number of chair legs. Cluster A is formed according to the straight angle chair legs; cluster B according to the circular base of chair legs; cluster C according to the widening angle of chair legs; cluster D according to the box-form of chair legs, and cluster E according to the rounded form of the chair legs. In each cluster, there are five sub-clusters based on the form of the backrests; (1) singular, (2) dual, (3) triple, (4) elongated and (5) semi-circular. According to designer C, there are two criteria for the formation of resemblance clusters; (1) backrests, and (2) chair legs. Primarily, Designer C formed 15 resemblance clusters according to the backrests of the chairs and within these resemblance clusters Designer C formed sub-clusters based on chair legs. (Fig.3). The resemblance relations are structured on the common/repeating elements. For instance, similar chair-legs form a resemblance cluster for Designer A and chair-leg forms and number for Designer B. For Designer C, backrests form a resemblance cluster. (Fig.4). The visually distinguishing criteria that form resemblance clusters are different, but the identification of common features that causes resemblance is the same tendency to form a resemblance cluster. Therefore, as expected, the formation of resemblance clusters, accordingly the build-up of visual structures, actualized in a subjective and designer-dependent manner.
Figure 3: Perception-Based Structurings of Designer A-B-C

Figure 4: The Repetition of Similar Textures

<table>
<thead>
<tr>
<th>Designer A</th>
<th>1</th>
<th>2</th>
<th>16</th>
<th>47</th>
<th>34</th>
<th>42</th>
<th>43</th>
<th>66</th>
<th>68</th>
<th>78</th>
<th>79</th>
<th>96</th>
<th>64</th>
<th>28</th>
</tr>
</thead>
<tbody>
<tr>
<td>Designer B</td>
<td></td>
<td></td>
<td>99</td>
<td>10</td>
<td>24</td>
<td>74</td>
<td>75</td>
<td>50</td>
<td>106</td>
<td>68</td>
<td>73</td>
<td>104</td>
<td>58</td>
<td>65</td>
</tr>
<tr>
<td>Designer C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Retrieval-Based Structuring

Retrieval-based structuring is based on the exploitation of 3D object retrieval methods for large design search spaces. It is an objective filtration process and has the potential to identify similarities between design instances after the generation process. It requires the automated build-up of visual structures.

3D object retrieval methods are geometry-based search algorithms that are used for the evaluation of similarity and the classification of 3D objects based on benchmarking [22]. 3D object search requires the shape descriptor which defines and represents the features and information of an object, and a 3D object repository that provides a benchmarking scheme [22]. In the 3D object retrieval research, benchmarking schemes are used for the evaluation of the retrieval algorithms [26]. The evaluation is based on testing and comparison of the search algorithms and provides feedback for 3D objects retrieval methods. In general terms, similarity evaluation is conducted by matching the geometries and geometric features of compared objects in a benchmark scheme.

3D object retrieval is based on four phases: (1) query formation, (2) feature extraction, (3) dissimilarity computation and (4) retrieval [23]. Query formation is the selection of the 3D object to be compared; feature extraction is the determination of the features of the selected object; dissimilarity computation is based on the comparison of the object in a 3D object repository; and the retrieval phase is based on the detection of the objects that have the lowest dissimilarity value obtained in the third phase [24].

Within the context of visual structures, the 3D benchmarking process may be used at the end of the design synthesis process; the design search space can be considered as a 3D model repository, a benchmarking scheme. The designer identifies the reference object that is used for
benchmarking. The methods to generate/identify the reference object can be a subject of another research. It is assumed that there is a reference object that is identified/generated by the designer. The reference object can be used to query the visually similar design instances. Also, in correspondence with type-instance discussion, the reference object is the type that defines the group of similar design instances. Furthermore, the identification of the reference object becomes the filtration criteria through the benchmarking process. After the identification of the reference object, the retrieval process carries out the comparative evaluation between design instances in a design search space. With such evaluation, the 3D retrieval algorithm identifies similar design instances. (Fig.5). Therefore, resemblance clusters are formed accordingly in an automated fashion. The retrieval-based visual structuring process is automated, and the generation of resemblance clusters is not as subjective as perception-based visual structuring. From this point, such structures challenge the designer’s evaluation and subjectivity throughout the formation of resemblance clusters. However, retrieval-based visual structuring facilitates designer interaction with complex design search spaces by structuring the design search space. Furthermore, there are developing 3D object retrieval methods that simulate the designerly actions of classifications and detection of similarity.

**Optimality-Based Structuring**

Optimality-based structuring is an objective filtration strategy that is based on the optimality for large design search spaces. The visualization/analogue representation and clustering of design instances, which are located on fitness peaks in a fitness landscape are centric for such a structuring strategy. In comparison with the other two strategies, the relationships between the design instances are not structured in optimality-based visual structuring. Here, such structures result in a group of non-structured design instances and inform designers about formal properties and fitness states of the design instances. The most common method for visualization of GA is based on fitness evaluation of genomes illustrated by fitness-time graphs [25]. Sewall Wright suggests fitness landscapes as a visual metaphor to depict gene combination space and the evolution of a population [26]. Fitness landscapes illustrate the fitness status for a generated population according to the fitness criteria [27]. Wright depicts fitness landscapes with hills and valleys that generated populations move through [26]. Here, the optimization process associates with the hill-climbing metaphor that signifies the movement of a population through regions of low fitness to high fitness; a random walk through a surface in a three-dimensional space. Fitness landscapes can visualize high-dimensional search spaces by low-dimensional representations, with fitness landscape visualization limited to maximum two dimensions [28]. Therefore, more than two degrees challenge the visualization of search spaces. However, with the developments in current data visualization technologies, new methods for complex search space visualization models based on dimension reduction are being developed. One of these developing methods is proposed by McCandlish and is based on the segregation of the genomes located at different fitness peaks [28]. By segregating the genomes to different peaks, the major features of the peak can be displayed graphically [28]. Here, McCandlish (2011) proposes a symbolic representation of design instances at the same peak. For optimality-based visual structuring, McCandlish’s method can be used as long as the design instances visualized with analogue representation. The segregation of design instances that are located on different peaks automatically structures design search space according to the fitness scores of design instances. The identification and visualization of these design instances may inform the designer about formal features of design instances that belong to the same peak. This facilitates the visual interaction and guidance of a designer through the design search space. Furthermore, due to convergence in the optimization processes, design instances at the same peak are visually similar. Such similarities have the potential to form resemblance clusters after the generation phase. The determination of major features of fitness regions corresponds with the type.
As optimality-based visual structuring strategy demonstrates objective filtration based on fitness scores, such structuring may pose a challenge for the subjectivity of designerly evaluation. However, these structures reduce the complexity in a design search space by limiting the number of design instances with the number of the design instances that are located at the peaks. This reduction facilitates the designerly interaction with refined and smaller design search spaces.

**FINDINGS AND RESULTS**

There are several remarks that have to be highlighted and concluded. Primarily, the perception of a similar feature in a design set is defined as resemblance. Resemblance in a design space can be used as an operational method to build visual structures.

Secondly, according to the case study, each designer built a different structure using the same design instances according to different resemblance criteria for grouping. This study highlights the subjectivity of the designer and designerly evaluation.

Thirdly, there is a need to compare and contrast the visual structuring methods in brief. In perception-based, the designer is directly in dialogue with a complete set of design instances. In comparison with perception-based structuring strategy, the designer is in dialogue with filtered and structured design search spaces. Here, in perception-based, the designer is involved through the whole process of visual structuring; therefore, the visual structuring process includes a subjective component. However, in retrieval-based and optimality-based structuring, the designer does become involved through the automated visual structuring. Therefore, subjectivity may decrease and is not centric in these strategies. Despite the subjectivity states of such strategies, both of them propose refined and structured design search spaces with which the designer can easily interact. Accordingly, these structuring methods provide visual guidance to the designer and facilitate the designerly interaction and evaluation after the design synthesis.

**IMPLICATIONS FOR THE FURTHER RESEARCH AND CONCLUSION**

For further research, all structuring strategies hold the potential for different software prototype implementations as a proof of concept within the scope of information modelling.

For perception-based structuring, although it requires a manual build-up, a filtration software that reduces the number of design instances can be implemented. Furthermore, a platform, a software with a visually rich and supportive interface that contains a whole set of visual representations of design instances, may enable designers to structure their resemblance clusters literally as a tree-structures as they are mapped above.

For retrieval-based structuring, a software that integrates 3D benchmarking algorithms in digital design synthesis has the potential to be prototyped and implemented for the further improvements.

For optimality-based structuring, regarding the current developments in data visualization technologies, a software is required that can be implemented as a continuation of the fitness peak identification within the framework of McCandlish’s method, one that visualizes the design instances as analogue representations, and has the potential for facilitated integration of the designer within the fitness landscape, without even interacting with the symbolic representations of the peaks.

If the potential software prototypes are to be implemented, there should be another research conducted to analyse the integration of the designer through the design synthesis process via those prototypes.

**REFERENCES**


<table>
<thead>
<tr>
<th>Author</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abbas, Günsu Merin</td>
<td>185</td>
</tr>
<tr>
<td>Balla-S. Béla, Szilvia</td>
<td>105</td>
</tr>
<tr>
<td>Bertin, Vito</td>
<td>79</td>
</tr>
<tr>
<td>Botzheim, Bálint</td>
<td>213</td>
</tr>
<tr>
<td>Bödő, Gábor</td>
<td>235</td>
</tr>
<tr>
<td>Castellon Gonzalez, Juan José</td>
<td>177</td>
</tr>
<tr>
<td>Chang, Tengwen</td>
<td>163</td>
</tr>
<tr>
<td>Chaszar, Andre</td>
<td>227</td>
</tr>
<tr>
<td>D’Acunto, Pierluigi</td>
<td>177</td>
</tr>
<tr>
<td>Datta, Sambit</td>
<td>163</td>
</tr>
<tr>
<td>De Luca, Francesco</td>
<td>195</td>
</tr>
<tr>
<td>De Paris, Sabine</td>
<td>55</td>
</tr>
<tr>
<td>Dino, Ipek Gürsel</td>
<td>185</td>
</tr>
<tr>
<td>Dumitrescu, Delia</td>
<td>203</td>
</tr>
<tr>
<td>Elkady, Shawkat L.</td>
<td>169</td>
</tr>
<tr>
<td>Ezzat, Mohammed</td>
<td>111</td>
</tr>
<tr>
<td>Fehér, András</td>
<td>235</td>
</tr>
<tr>
<td>Fricker, Pia</td>
<td>119</td>
</tr>
<tr>
<td>Füzes, Bálint Péter</td>
<td>73</td>
</tr>
<tr>
<td>Gidófalvy, Kittí</td>
<td>213</td>
</tr>
<tr>
<td>Gyulai, Attila</td>
<td>67</td>
</tr>
<tr>
<td>Hadzijianisz, Konsztantinosz</td>
<td>235</td>
</tr>
<tr>
<td>Hegyi, Dezső</td>
<td>73</td>
</tr>
<tr>
<td>Heinrich, Benjamin</td>
<td>243</td>
</tr>
<tr>
<td>Iványi, Péter</td>
<td>221</td>
</tr>
<tr>
<td>Kari, Szabolcs</td>
<td>67</td>
</tr>
<tr>
<td>Kikunaga, Patricia Emy</td>
<td>213</td>
</tr>
<tr>
<td>Koenig, Reinhard</td>
<td>15</td>
</tr>
<tr>
<td>Kolarevic, Branko</td>
<td>27</td>
</tr>
<tr>
<td>Kulcke, Matthias</td>
<td>61</td>
</tr>
<tr>
<td>Lam, Wai Yin</td>
<td>79</td>
</tr>
<tr>
<td>Lellei, László</td>
<td>67</td>
</tr>
<tr>
<td>Lorenzo, Wolfgang E.</td>
<td>249</td>
</tr>
<tr>
<td>Lovas, Réka</td>
<td>235</td>
</tr>
<tr>
<td>Lucchi, Elena</td>
<td>155</td>
</tr>
<tr>
<td>Matsubayashi, Michio</td>
<td>87</td>
</tr>
<tr>
<td>Nováková, Kateřina</td>
<td>133</td>
</tr>
<tr>
<td>Nuno Lacerda Lopes, Carlos</td>
<td>55</td>
</tr>
<tr>
<td>Pascucci, Michela</td>
<td>155</td>
</tr>
<tr>
<td>Pletenac, Lidija</td>
<td>141</td>
</tr>
<tr>
<td>Reffat M., Rabee</td>
<td>169</td>
</tr>
<tr>
<td>Reith, András</td>
<td>213</td>
</tr>
<tr>
<td>Riedel, Miklós Márton</td>
<td>67</td>
</tr>
<tr>
<td>Rossado Espinoza, Verónica Paola</td>
<td>127</td>
</tr>
<tr>
<td>Sajtos, István</td>
<td>149</td>
</tr>
<tr>
<td>Sárközi, Réka</td>
<td>221</td>
</tr>
<tr>
<td>Schmitt, Gerhard</td>
<td>15</td>
</tr>
<tr>
<td>Seddik, Moamen M.</td>
<td>169</td>
</tr>
<tr>
<td>Selvær, Harald</td>
<td>99</td>
</tr>
<tr>
<td>Sik, András</td>
<td>67</td>
</tr>
<tr>
<td>Smolik, Andrei</td>
<td>163</td>
</tr>
<tr>
<td>Strommer, László</td>
<td>49</td>
</tr>
<tr>
<td>Sundfør, Ingolf</td>
<td>99</td>
</tr>
<tr>
<td>Surina, Dóra</td>
<td>235</td>
</tr>
<tr>
<td>Szabó, Beatrix</td>
<td>235</td>
</tr>
<tr>
<td>Széll, Attila Béla</td>
<td>221</td>
</tr>
<tr>
<td>Szilvás-Nagy, Márta</td>
<td>105</td>
</tr>
<tr>
<td>Szollár, András</td>
<td>213</td>
</tr>
<tr>
<td>Ther, Tamás</td>
<td>149</td>
</tr>
<tr>
<td>Vári, Barnabás</td>
<td>235</td>
</tr>
<tr>
<td>Watanabe, Shun</td>
<td>41, 87</td>
</tr>
<tr>
<td>Wurzer, Gabriel</td>
<td>243</td>
</tr>
<tr>
<td>Xu, Lei</td>
<td>93</td>
</tr>
<tr>
<td>Yajima, Kazumi</td>
<td>33</td>
</tr>
</tbody>
</table>
The aim of these workshops and conference is to help transfer and spread newly appearing design technologies, educational methods and digital modelling supported by information technology in architecture. By organizing a workshop with a conference, we would like to close the distance between practice and theory.

Architects who keep up with the new designs demanded by the building industry will remain at the forefront of the design process in our information-technology based world. Being familiar with the tools available for simulations and early phase models will enable architects to lead the process. We can get “back to command”.

The other message of our slogan is <Back to command>. In the expanding world of IT applications there is a need for the ready change of preliminary models by using parameters and scripts. These approaches retrieve the feeling of command-oriented systems, although, with much greater effectiveness.

Why CAADence in architecture?

"The cadence is perhaps one of the most unusual elements of classical music, an indispensable addition to an orchestra-accompanied concerto that, though ubiquitous, can take a wide variety of forms. By definition, a cadence is a solo that precedes a closing formula, in which the soloist plays a series of personally selected or invented musical phrases, interspersed with previously played themes – in short, a free ground for virtuosic improvisation."