06_Iordanova 30/01/08 10:19 am Page 685

Teaching Digital Design Exploration: Form Follows…

Ivanka Iordanova

Teaching Digital Design Exploration: Form Follows…

Ivanka Iordanova

This paper presents some challenges of teaching computational geometry to architectural students, and proposes a multi-level pedagogical scheme introducing associative geometry and parametric modeling/design into architectural design education. It reports on two pedagogical experiences: one held in the context of a spatial geometry course in the first year of education; and another one, in a digital design studio with third-year architectural students. More specifically, it discusses the impact on design exploration of *a library of interactive referents models* introduced into the architectural studio. Situated in the 'performance' paradigm of digital design methods, they allow for design object explorations based on modification of architecturally meaningful features (structural, environmental, functional, etc.). The form of a design object can thus 'follow' function, structure, or even sustainability. The digital methods and the design knowledge transferred by the interactive models, together with their visual nature, are found to amplify the processes of 'seeing-as' and the 'reflective conversation with the situation' considered essential for creative design.

1. INTRODUCTION

New architectural tectonics, a product of digital design thinking, has started animating some architectural landscapes. What was until recently considered only possible in virtual space is now sending 'representatives' into the real world. Design theory is slowly changing in order to reflect these transformations and accommodate cognitive processes interacting with computational ones. Design teaching seems to be the last realm to enter into a process of change and to add to its traditional approaches new ones, preparing the students for digital ways of creating architecture [1].

Design exploration contributes significantly to the emergence of new ideas. Creating variations and generating multiple ideas (indicating divergent thinking) are reported to be an important prerequisite for quality and creativity in architectural design [2, 3]. At the same time, traditional use of computer in the design studio is mostly for representing the project, infrequently for communicating the process of its making, and almost never for assisting generation of design ideas.

The aim of the research work presented in this paper is to find out how to teach computational methods in the context of a design studio and what are the prerequisites for this. Once solutions to these questions are proposed, we will determine whether these digital approaches modify the processes of 'seeing-as' and 'reflective conversation with the situation' during architectural design, as some authors state [1], and if so, in which way?

This article deals with some of the challenges of teaching associative geometry and parametric design to architectural students, and proposes a multi-level pedagogical scheme introducing computational geometry (CG) into architectural design education. It also analyses the impact on design exploration of a library of interactive referents introduced into a digital architectural studio.

The article is structured in the following way: We will first draw a schematic picture of the use of CG in architectural design and will situate our approach on this background. Then, we will discuss some challenges of teaching geometry and programming to future architects and we will propose a 'joint strategy' for this. Teaching digital design exploration in the context of the studio project will be presented next. In this context, a library of referents including *interactive chunks of knowledge* will be studied for its role in design learning. The principle of '*form follows function*', or '*structure*' or even '*sustainability*' will be exposed. 'Micro observations' in the design studio will be presented and some results from the validation of this approach will be shown and discussed.

2. ON COMPUTATIONAL GEOMETRY AND ARCHITECTURAL DESIGN

Since the introduction of computer into their practice, architects are directly or indirectly served by computational geometry in order to create

3D forms and spaces. A simple line "drawn" on the screen, the surface of a pitched roof, the volume of a dome, the representation of a soft shell-like form, the simulation of a kinetic structure, or a stress diagram are all created by the methods of CG. Many aspects of such digital representations can vary, for example, size, proportions, relations between parts, way of generation of a form, type of visualization (or output), etc. But in order to be able to take advantage of these possibilities and use them for design exploration, as suggested by Glanville [4], they should have been taken into account in the digital description of the object by parametric modeling.

2.1. Parametric Modeling and Design

A building, which normally responds to numerous requirements and conditions, seems to be a perfect object for a parametric description that considers relations, rules and procedures. But the complexity of architecture is such that any model taking into account many of its aspects becomes either simplistic or too deterministic, making interaction with the design object very difficult and design exploration–impossible. As a consequence, purely formal parameterizations are most frequent.

According to Hernandez [5], "parametric design is the process of designing in environment where design variations are effortless, thus replacing singularity with multiplicity in the design process". But the possible variations can be quite different depending on the attribute which is parameterized (dimensions, relations or generation algorithms). Consequently, we find that the term 'parametric design' is not precise enough as its computational bases or its architectural meaning may vary. In this paper we will specify each particular context for a better understanding.

A specific kind of parametric modeling, the 'modeling of actions' that can produce a digital model aiming at particular objectives, is proposed by Tidafi [6]. Burry [7] identifies the principles of parametric design. De Paoli [8] speaks of a 'procedural model' and envisages its use in design. Hernandez [5] calls a similar method 'design procedure'. It is defined as "a set of instructions that performs actions that generate parameterized geometrical models". In our opinion, this is the kind of parametric modeling which is best suited for the description and transfer of architectural know-how, as well as for the exploration of design. The architectural meaning of the parameterized attribute can determine the kind of model to be produced and the possible interactions with it: modifying directly the form, influencing the form through changes in the structural characteristics, reflecting transformations of forces from the physical environment (sun, wind, sound, traffic, etc.).

2.2. Paradigmatic Classes of Digital Design Models

Oxman [1, 9] combines come computational and architectural aspects and identifies four paradigmatic classes of digital models: CAD (supporting posteriori automation of design drawings and visual models), formation (exploiting different geometries and parameterization), generation (considering shapes as a result of pre-formulated generative processes), and performance (which might rely on formation and generation processes, but includes the influence of external forces like structural loads, acoustics, site, etc). The last three paradigms could enrich the design methods of an architect with methods made possible by computers. All these have parametric modeling and algorithms as a creation basis. The 'performance' paradigm has been the one in which we place our research work for almost a decade now. It has been oriented to domain knowledge and design know-how modeling, and to their reuse in new designs. This is also the ultimate objective of our design-studio approach described later in this article.

Parametric design is characterized by generative processes related to movement, and new tectonic elements emerge through it [10]. High-end computer software is aiming at providing architects with the possibility of creating and exploring models of design objects through the processes of formation, generation and performance (Generative components, for example). The use of this kind of software is not easy, though, especially for students entering the school of architecture with different backgrounds. The scientific bases of CG, geometry and programming, should be taught to them beforehand. Our approach introducing these concepts and knowledge early into the curriculum is discussed next.

3. ON SOME CHALLENGES OF TEACHING GEOMETRY AND PROGRAMMING TO FUTURE ARCHITECTS

The present context of a school of architecture is not always favorable to geometry or programming classes. In the "Beaux arts" tradition, architecture is much more an art and philosophy than a science or something reasoned.

3.1. Teaching Geometry

With the use of computer visualization, descriptive geometry lost its unbothered position in the curriculum; and mathematics and geometry courses started being considered obsolete.

However, the role of geometry for amplifying spatial perception and imagination, as well as the role of mathematics in developing logics and reflective process are well known [11, 12, 13]. There are international discussions engaged on these issues (in the Nexus Network Journal, for example). Some fundamental questions on the objectives of architectural education in general are being asked: For instance, should it prepare for *one profession*, or should it provide a methodological-cultural background that allows for future flexibility [14]?

The content of geometry courses is being reconsidered in light of the latest requirements to the architectural profession. Two main objectives are being pursued: developing spatial thinking and imagination; and preparing for an informed and active use of computational geometry software, or even for 'toolmakers' [15].

3.2. Teaching Programming

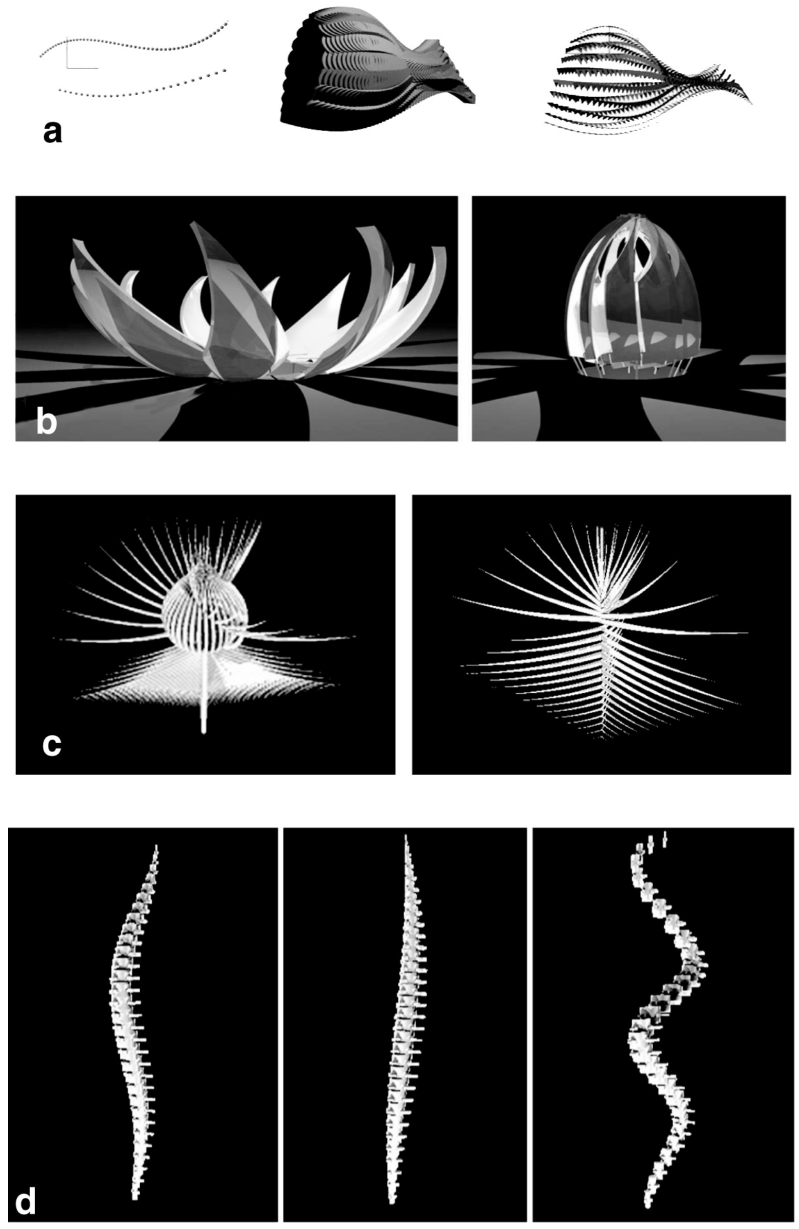
According to Terzidis [16], "It has become unavoidable to enter into the black box of programming in order to make a truly creative use of the computer". This is particularly true for 'digital' architects like Greg Lynn or Norman Foster who generate their ideas with the aid of computer generation. This statement is less valid for architects like Frank Gehry (or his team) who use computer programs for the realization of their essentially sculptural ideas [17]. Some researchers report the use of programming learning as a means for extending spatial and temporal understanding of students [18]; as well as its impact on the personal design-process elaboration [19]. Dearden even tries to find indications for a 'reflective conversation with the materials of the situation' by studying the programming style and notations of architects [20]. Thus, we consider necessary teaching low-end programming early in the curriculum in order to prevent, as much as possible, the creation of preconceptions about the nature of design process.

3.3. A Joint Approach

When dealing with a problem, Vigotsky [21] recommends: "Always connect, never isolate". This way, in our opinion, teaching spatial geometry and programming should be connected to real architectural problems in order to be well understood and accepted by students. In a proposed joint approach, they are part of different types of representation techniques taught to students in an obligatory, at least for now, first-year Spatial Geometry course. The three modeling (or representation) strategies explained and applied to architectural context are inspired by the book "Nature, Geometry, Architecture" [22] and consist of:

1) Direct description of the geometry of an object.
2) Structural description of a surface or volume: through trajectory and generating line, for example.
3) Generative description: requiring a source-figure and an algorithm for its transformation and movement in 3D space, for example.

After having worked with associative geometry software during the term, students were asked to explore a morphogenetic form in their final project and to link nature-principles to architectural structure and form. They had to define a strategy for representing the identified generating principles of the form, to program it (using PovRay) and to generate at least two different variations that could serve as new designs based on the modeled principles. Some examples of students' work can be seen on Figure 1.



► Figure 1. Examples of models generated by the first year students: a: Gehry's 'fish': Isabelle Gallant and Karine Legault Leblanc. b: Baha'i Church: Gabriel Kakon and Jessica Lee Owens. c: A shell: Andy Oshima and Jean-Rémy Thomas. d: Spinal column: Maxime Auger-Daoust and Anne-Sophie Gauvin.

Despite the inevitable difficulties and the abundant help provided by teacher and assistant, we believe the results of this cohort of 85 students to be quite encouraging. Compared to previous experience with teaching programming and scripting to masters level students, we found less preconceptions and greater openness to methods different from traditional ones. We can not evaluate, for now, the effect that this approach in the first-year geometry course will have on digital design learning, because the first cohort of students that have studied modeling strategies is not yet in their third year of education (when the digital studio takes place).

4. TEACHING DIGITAL DESIGN EXPLORATION

Bringing exploration of digital design into the architectural studio is the long-term objective of the approach described just above. In our curriculum, the first occasion to teach digital design methods comes forth during our 'experimental studio', at the beginning of the third year. Despite its name, this studio tries to apply the exploration of digital models to a real architectural project assigned at the beginning of the term. Hence, relation to the site, climate, cultural context, etc. are all taken into consideration. This is a major difference to other studios where generative methods are taught either free of context, or the context is found once the form has been created [23, 9].

Given the visual and tacit architectural culture which is well embraced by the students during the first two years at the School of Architecture, it is difficult to imagine teaching programming in a straightforward way, even if it were for design exploration. A more architect-friendly strategy was proposed for this: a *library of referents* containing, among other forms of representation, *interactive digital models* generated on the basis of domain knowledge and explicit design process. We use the term referent to indicate all kinds of objects or phenomena to which an architect would refer, either for inspiration or for finding 'howto' information, during the design process.

4.1. The Referents

Referents provide holistic solutions to complex situations and are frequently used for the resolution of ill-defined problems [24, 15]. Much research work has been carried out on the role of precedents and metaphors in design and design-learning, as well as on their organization into conceptual networks [25, 26, 27, 28, 29]. The referents proposed in this paper differ in their content (encapsulating domain knowledge and generating-process information) and in the form of their digital representation (including *interactive models of chunks of knowledge*).

Theoretical Bases

The definition and the introduction of the referents library have already been the subject of other communications, so we will present them quite briefly here. Their theoretical bases are:

- The largely accepted use of referents during the design process (as source of inspiration, or as carriers of implicit architectural knowledge and know-how), identified by design theory research.
- The fragmented and dynamic organization of memories in which 'chunks of knowledge' represent attributes or features of the mental representation of an object.

- The multi-format representation of referents for a better cognitive interaction with them.
- The need for explicitation of the generating processes when referring to metaphors or precedents, with the purpose of their possible reuse in new design situations.
- The possibility to create conditions for design exploration based on variations in some architecturally meaningful parameters like structural loads, light, temperature, wind, acoustics, relations to the site, or cultural context.
- The potential provided by computing methods for encapsulating architectural knowledge and processes through parameterization, algorithmic description, etc.

Knowledge Transfer

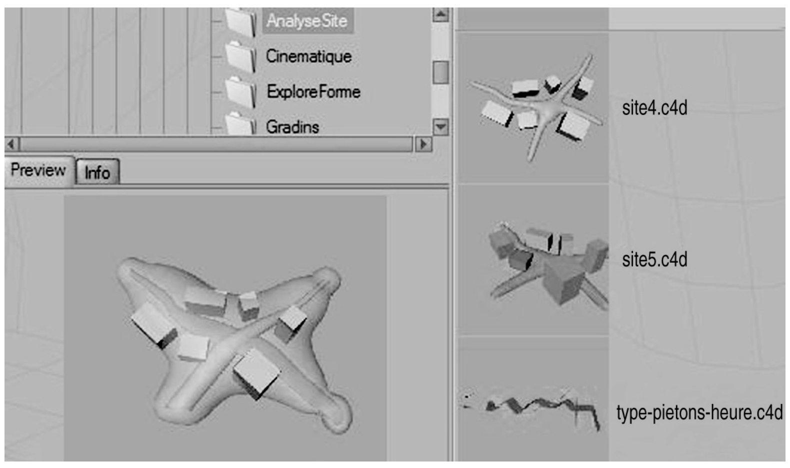
A special attention in this list is paid to encapsulation of architectural knowledge and the role that it might play in design exploration. Nowadays architectural culture is extremely visual, taking advantage of the powers of a 'visual thinking' and, in the same time, often suffering from lack of information on the 'how' and 'why' of a drawing or picture. At the same time, some authors consider that young architect lack architectural knowledge, blaming partially the computer for this [26, 30].

From an educational point of view, using algorithms gives the possibility of encoding architectural knowledge linked to rules and laws (structural, climatic, compositional, etc.). It also enables to encapsulate processes, to visualize and test them as simulations in time (energy optimization, manner of production, etc.). Thus, the introduction of referents library in the digital studio pursues a double objective: educating students in a new way of design thinking and exploration (based on process rather than on result); as well as providing architectural know-how linked to design process and building performance. This way, together with purely formal explorations of an object of design, a student is provided with the possibility of influencing the form through manipulation of non-geometric architecturally meaningful information. In this respect, we could say that *form could follow function, or structure or even sustainability* [31].

Implementation of the Referents Library

The referents are represented in the library by information of different formats (pictures; textual information; *interactive digital models of chunks of knowledge*; video films on the creation of the models and their possible transformations; and semantic annotations). Technically, each *interactive digital model* is a file in the format of the modeling software which is used (Cinema 4D in the case of the tested prototype). An algorithmic generation of some of these models allows for the description of architectural know-how embodied in them (compositional rules, acoustic formulae, visibility optimization, structural dependencies, bioclimatic laws, etc.). The participating parameters are named according to their architectural meaning (for example: height of row, number of people in the audience, maximum day temperature, etc.) and this contributes to a better understanding of the model. Interactivity is provided by modifying the parameters and/or the algorithm. One referent (a famous building, for example) can have many interactive models linked to it, depicting different aspects of architectural know-how having contributed to its design. From a cognitive point of view, the representation of chunks of knowledge of a referent by means of an interactive digital model includes several different knowledge formats: visual information and some rules underlying it; some aspects of propositional knowledge provided by annotation or simply by naming the parameters participating in the rules or processes; and finally procedural knowledge. The latter can be comprehended by a student in three ways: by observing the model's behavior when parameters vary, by looking at the description of the rule or algorithm, and by watching a video-recording explaining and showing the creation of the model.

Ideally, the referents library should be software-independent and shared on the Web or in another accessible network environment. For the time being, the prototype library used for experimentation of the interactive models concept is accessible via the interface of a modeling software program (Cinema 4D). It was chosen because of the relatively good learning curve (the students were exploring the software together with the work on their architectural project in the studio), the graphical programming interface it provides, and the user-friendly browser it offers (Figure 2). The referents are organized in folders according to their architectural meaning (site analysis, structural, acoustic, wind, etc.), and ***not*** by the format of the information in them (text, model, picture). The structure is open and flexible, thus allowing the students to customize it



◀ Figure 2. A snapshot of the referents library as it appears in the browser. The *interactive chunks of knowledge* shown here are linked to site analysis and exploration, and are inspired by the 'Information Architecture' paradigm.

on their own computers. During the term, the referents library was constantly enriched by new interactive digital models created by students (after approval by their tutor).

With the purpose of an optimal comprehension of the referents library and of the design methods it proposes, an appropriate teaching approach had to be defined.

4.2. The Teaching Method

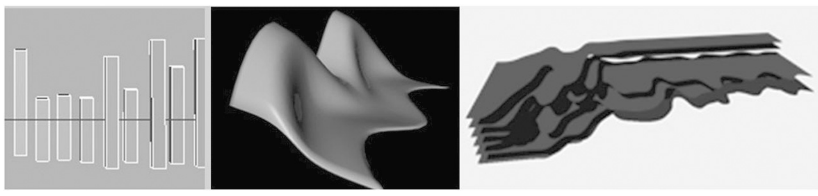
Digital design teaching makes the object of a growing number of scientific publications and communications [1, 32]. Unlike real architectural practice, a teaching method put forward by Oxman [9] proposes to students a laboratory for experimentation and, only at the end of the process, finds a suitable situation for the designed object. Our objective, though, was to work on a real project, but with digital means. In the fall of 2006, the term project was the design of the Home of the City's Symphonic Orchestra. The teaching method used in the design studio, was inspired by the 'preparatory classes' of VHUTEMAS [33] and Bauhaus [34]. Thus, propaedeutic exercises were proposed to students, together with the introduction of the corresponding modules of the referents library. The interactive chunks of knowledge from the library play a *double pedagogical role*: in terms of *design knowledge transfer;* and as help to students for a better *performance in their 'proximal zone of learning'* [21]. This is the 'zone' in which a student can not perform given tasks on his/her own, but is capable of understanding them and achieving them successfully with external help. This process enlarges the learning capacity and the speed of learning. Observations were held in the design studio in order to evaluate the eventual effect of the use of the referents library on the students' design process.

5. OBSERVATIONS IN THE DESIGN STUDIO

The research reported in this part of the article was already subject of other communications. A qualitative methodology (case study and grounded theory) is used in it. Two kinds of observations were held with the purpose of validating the proposed pedagogical approach for teaching CG for design exploration: a longitudinal one during the whole term; and short in-depth micro observations. A third year digital design studio has served as a field of observation for several years now. Only results from the last studio will be discussed here, but the previous ones served as basis for the development of the teaching approach and of the referents library.

5.1. The Studio Project

Design exploration was a main requirement during the work on the studio project. Students were asked to generate at least two different concepts for their projects and to propose different '*partis*' for each of them. An interesting effect of using the *interactive chunks of knowledge* from the



◀ Figure 3. Digital exploration of an object of design based on an interactive model influenced by sound-frequency: Left: the model provided in the library. Middle and right: Variations of this model generated by students.

referents library, or other parametrical functions, was to create one model for each concept and then to "play" with it in order to come to many variances (Figure 3). After having completed a questionnaire, some students mentioned that they were somewhat surprised and excited about these new perspectives on architectural design.

Design ideas frequently emerged during such explorations. It was not easy, though, to keep the exploration possibility all through the design development process. Special discipline was required for this. In certain cases, because of the long calculation time, the students were obliged to keep one model for exploration and to work with a derived one for the 'materialization' of the building (Figure 4).

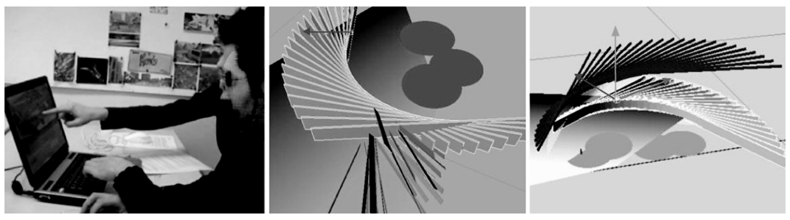
◀ Figure 4. Form generation influenced by music (work of Jerome Taillandier). Left: Exploration based on a modeled referent. Middle: Idea for the project. Right: The final project.

When asked, students mentioned the positive role that the referents library had played as a source of design knowledge and assistance in the elaboration of their own design strategy.

5.2. Micro Observations

In order to observe with greater depth the students' design processes, we performed micro-observations in the 11th week of the term. The observation protocol was based on spontaneous verbalization (dialogue) while the 10 students (5 male and 5 female) worked in teams of two (Figure 5, left). Each team had to work on one computer, but with two mice for better interaction with the digital model and environment. The conceptual design task was a summer theater with exhibition space. During the 2h30 given for the completion of the small project, the design sessions were recorded by video-cameras and by a screen-capturing software. Conversations, design actions and gestures from the recorded data were then synthetically transcribed and coded according to themes of interest: design moves, referents library use, emergence of ideas, etc. Qualitative comparison was then made with previous observed data processed in a similar way.

General observation of the digital design methods used showed that parametric generation of design was successfully exploited by three (of the five) teams. Two of these teams were working exclusively parametrically (Figure 5, middle and right). The remaining two teams were trying to use parametric forms and distributions in the context of their architectural task, but without much success. Almost all of the students (9) used digital design exploration during their work on the term project, though. The time constraints might be responsible for this difference in their strategies.



▶ Figure 5. Left: A team of two students during the micro-observation. Middle and right: Interactive digital exploration of the design object, using parametric modeling.

Aside from research interest in micro observation, pedagogical objectives were perceived with it as well. Intensive work, exchange of experience and ideas were highly appreciated by the students.

5.3. Results

The collected data is a precious source of information on many facets of the design learning process. Several phenomena were particularly observed during the micro observations: the emergence of design ideas; the reuse of referent models knowledge; autonomous parametric modeling; moments of seeing-as; and moments of 'reflective conversation' (Schön [35]).

Design Ideas Emergence

Only major design ideas were observed. For an idea to be identified as such, both naming and explicitation of its relation (or contribution) to the project had to be available in the recorded data. As the 'quality' of an idea is very difficult to judge, criteria for a 'major idea' was (1) its pursuit till the end of the project, or (2) its substantial exploration before being abandoned. Moreover, we have seen earlier in this article that cognitive studies on creativity considered the generation of a (reasonably) great number of ideas as a condition for having some good ones among them. In this respect, quantity might be seen as an indicator of potentially good quality.

Two main 'emergence patterns' were identified: (1) general definition of the project *from the very beginning* of the design period, after which only minor ideas emerged while the 'objective' was the representation of the 'project idea'; and (2) identification of the leading principles of the project, after which the design was *guided by an exploration process*. The first 'pattern' was noticed in the teams not working parametrically. They generated fewer ideas (6 for each of the teams) and mainly at the beginning of the design session. The second one was present in the teams using parametric digital methods, and generated respectively 9, 12 and 14 ideas for the three teams.

Referent Models Knowledge Reuse or Direct Parameterization

In the reported experiments, three stages of use of the modeled referents were observed: (1) Using models with difficulty, wanting to see the video-film for help; (2) Using models with ease, modifying, and transferring to other situations; (3) Independent use of the rules behind the model and autonomous creation of new models.

The teams that never succeeded to include any type of parametric exploration in their micro-projects, only tried once to use an interactive referent model from the library. Among the other three teams, one used only autonomous parameterization (3 different kinds), a second one only used interactive referent models (9 times), and the third one referred to both.

An acknowledged risk of referring to precedents is copying or imitation. While the purpose of the encoded generating principles of a referent is to be learned and reused, its formal characteristics should have a considerable degree of novelty compared to the source. The originality of the projects having integrated generating principles from modeled references was considered adequate to an extent that no formal similarity could be identified in some of the cases (Figure 3). When transferring scientific knowledge (visibility, acoustics), though, the link was easily identifiable, as some formal aspects of the newly designed object remained the same as the referent.

Moments of 'Seeing-as' and 'Reflective Conversation with the Situation'

According to Goldschmidt [24], the design process relies a lot on visual thinking, including a cycle of 'seeing-that' and 'seeing-as', using visual representation of the design object as a support. The 'seeing-as' indicates a creative leap in the project. We wanted to see whether digital exploration modified this cycle, and if the 'seeing-as' was still a prerequisite for emergence of a new idea. In a qualitative study like the one discussed here, no quantitative conclusions can be made, but a tendency can be seen. Thus, we noticed that the more moments of 'seeing-as' were found during the design session, the more numerous design ideas the teams have generated. Not all ideas, though, were generated as a result of a 'seeing-as' (19 such moments were identified in total).

Moments of 'reflective conversations' were acknowledged when reconsidering and reframing of the project was done, based on digital exploration or referring to previous experience. It was observed that the teams using parametric exploration had many more such moments than the others. An interesting discovery was that these moments were practically coinciding with the ones of 'seeing-as'. This seems to be a quite logical result, having in mind the complexity of a process of 'reflective conversation' which might provoke creative leaps.

5.4. Discussion

On the basis of the observations in the design studio, it is possible to evaluate positively the role of the referents library for the elaboration of the students' design. Some of the referent models played a role similar to the one attributed to metaphors – design inspiration; while others were transferring design knowledge as precedents are supposed to do. The double objective for the creation of the referents library is considered attainable, as on one hand, encoded know-how was transferred to the new object, and on the other, it was serving as a basis for digital design. This way, functional, structural and performative aspects of the future building are considered in the project development from the very beginning. The *form follows the variations of architecturally meaningful forces and attributes*: function, structure, sustainability, etc. As the referents library is accessible through a free modeling software, 'traditional' design strategies remain available as well, and can be combined with the interactive chunks of knowledge models where suitable and desired.

We found no contradiction to the 'seeing-as' and the 'reflective conversation with the situation' processes. Furthermore, we noticed that they occurred much more frequently when working with parametric models and when continuous explorations were done. This contradicts some hypotheses that continuous exploration obstructs creative leaps because of the lack of intermittent processes [36]. In fact, we were able to identify some kind of *rhythm in the digital exploration process*: first, students *generate a conceptual idea* – they decide which attribute or force to take into consideration; then, they *digitally explore* the conceptual idea, when unexpected forms can be created and *new 'ideas' generated*; an *evaluation process* determines whether to continue the design process or not. The 'control' of the architecturally meaningful aspect is in the hands of the architect, while the form can be surprising, as generated by computer. The process of 'seeing-as' can then occur, and give rise to new ideas – conceptual or formal ones. We consider this as a good level of human-computer symbiosis achieved thanks to teaching CG methods for design exploration.

The referents library played successfully its role as a *proximal zone learning catalyst*, as conceived. Its contents, though, has to be enriched in order to provide a more general collection of *chunks of knowledge*.

6. CONCLUSION

In this paper we proposed a multi-level pedagogical scheme introducing associative geometry and parametric design into architectural design education. We combined experiences different in context (geometry course and design studio), in time (first year or third year of education), and in length (the whole term or 2h30), in order to validate these innovative teaching methods and to study some of their cognitive effects.

Even if an explicit comparison and follow-up cannot yet be made (the cohort of the first-geometry course has not reached their third year yet), some positive effects can be identified in the knowledge acquired by students in associative geometry and programming, as well as in their attitude towards digital methods of design exploration.

Much more openness to the use of programming in architecture was observed in the first year students group, compared to the full of preconceptions group of masters' students. The third year students were quite curious, but their background was not adequate enough. This suggests the idea that innovative methods should have their place in the beginning of the curriculum when preconceptions are not too strong yet. The bases of computational geometry should be learned then, and thus, offer the basis of digital design to architectural students.

In respect to the proposed teaching approach in the design studio, we can conclude that algorithmically modeled interactive referents organized in a 'library' are very useful for digital design learning. They are most creatively used when internalized to a certain extent by students. From this moment on, they can offer the advantages of design exploration based on the knowledge coded in them, and thus, stimulate a process of 'reflective conversation with the situation', including new elements provided by the computer. The proposed referents library was estimated as a source of architectural knowledge and as valuable aid in digital design strategies. The library has to be constantly enriched by new categories and referents created by students, teachers, etc. Here, only the concept and its cognitive effects were tested, but in a future, the library could be shared on the Web and could provide links to different software programs, which we know, is not an easy task.

Acknowledgements

We thank the students from the AME-2129 Spatial geometry course for their inspired work, as well as the students from the ARC-3011 Digital studio for their participation in the observations. Special thanks to the professors from CAD Research Group of the School of Architecture at the Université de Montréal, as well as to Samir Benkheira who was a teaching assistant in the course and in the digital studio.

References

1. Oxman, R., Theory and design in the first digital age, *Design Studies* 2006, **17**(3): 229–265.
2. Osborn, A., *L'imagination constructive*. Paris, Dunod, 1965.
3. Csikszentmihalyi, M., The Domain of Creativity. *Theories of Creativity*. M. Runco and R. Albert. Newbury Park, Sage Publications**,** 1990, 190–214.
4. Glanville, R., Variety in Design. Variety in Design, Systems Research, 1994, 11(3), 95–103.

5. Hernandez, C., Thinking parametric design: introducing parametric Gaudi, *Design Studies*, 2006, 17(3): 309–324.
6. Tidafi, T., *Moyens pour la communication en architecture - Proposition de la modélisation d'actions pour la figuration architecturale*, PhD Thesis, Universite de Montreal, Montreal, 1996.
7. Burry, M., Paramorph: Anti-Accident. Methodology. *AD Profile 139 : Hypersurface Architecture II*, Wiley, 1999.
8. De Paoli, G., *Une nouvelle approche d'aide á la conception par ordinateur en architecture basée sur la modélisation d'opérateurs sémantiques et la création de maquettes procédurales*, PhD Thesis, Université de Montreal, Montreal, 1999.
9. Oxman, R., *Educating the Digital Design Thinker*. Communicating Space(s) [24th eCAADe Conference Proceedings], Volos, Greece, 2006, 198–205.
10. Liu, Y.-T. and Lim, C.-K., New tectonics: a preliminary framework involving classic and digital thinking. *Design Studies*, 2006, **17**(3): 267–307.
11. Baracs, J., Twelve Exercises on Spatial Perception, *Structural Topology*, 1988, **14**.
12. Baracs, J., The teaching of spatial geometry: the 9 basic problems. *Canadian Mathematical Society; Symposium "Mathematical Education: The teaching and learning of geometry: why, what, how."* Quebec, Canada, 2002.
13. Angulo, A., Davidson, R. and Vasquez, G., *Digital Visualization in the Teaching of Cognitive Visualisation*. ACADIA, 2001, 292–300.
14. Pedemonte, O., Mathematics for Architecture: Some European Experiences. *Nexus Network Journal*, 2001, 129–136.
15. Kalay, Y., *Architecture's New Media*, The MIT Press, 2004.
16. Terzidis, K., *Algorithmic Architecture*, Elsevier, 2006.
17. Abel, C., *Architecture, Technology and Process*. Boston, Elsevier, 2004.
18. Burry, M., Datta, S. and Anson, S., Introductory Computer Programming as a Means for Extending Spatial and Temporal Understanding. *ACADIA 2000: Eternity, Infinity and Virtuality*, Catholic University, Washington, DC, 2000.
19. Yakeley, M., *Digitally Mediated Design: Using Computer Programming to Develop a Personal Design Process*, PhD Thesis, Boston, MIT, 2000.
20. Dearden, A., Designing as a conversation with digital materials, *Design Studies*, 2006, **17**(3): 399–421.
21. Vygotsky, L. S., *Mind in Society*. Cambridge, MA, Harvard University Press, 1978.
22. Mickhailenko, V. and Kashtenko, A., *Priroda, geometria, architectura [Nature, geometry, architecture]*. Kiev, Budivelnik, 1988.
23. Weinand, Y., *New Modeling*, Presses polytechniques et universitaires romandes, 2004.
24. Goldschmidt, G., Visual Analogy - a Strategy for Design Reasoning and Learning, *Design knowing and Learning: Cognition in Design Education*, C. Eastman, M. McCracken and W. Newstetter. Amsterdam, Elsevier, 2001, 199–219.
25. Léglise, M., Des objets architecruraux aux objets de la conception architecturale, *Les objets en conception*, Paris, EUROPIA, 1998, 271–281.
26. Akin, Ö., Case Based Instruction Strategies in Architecture, *Design Studies*, 2002, **23**(4): 407–431.
27. Oxman, R., Precedents in design: a computational model for the organisation of precedent knowledge, *Design studies*, 1994, **15**(2): 141–157.
28. Heylighen, A. and Neuckermans, H., (Learning from Experience)? Promises, Problems and Side-effects of Case-Based Reasoning in Architectural Design, *International Journal of Architectural Computing*, 2003, 60–70.

29. Scaletsky, C., The Kaléidoscope System to Organize Architectural Design References, *IJAC*, 2004, **2**(3): 351–369.

30. Fernandez, P., Approches méthodologiques et modes opératoires dans le processus de conception architecturale, in M. Borillo and J.-P. Goulette eds., *Cognition et création: Explorations cognitives des processus de conception*, Sprimont, Belgique, Mardaga, 2002, 97–118.

31. Ábalos, I., Sustainable? CCA Study Centre – 2007 Colloquium. Montreal, 2007.

32. Pak, B., Ozener, O. and Erdem, A., Utilizing Customizable Generative Design Tools in Digital Design Studio: Xp-GEN experimental form generator, *IJAC*, 2006, **04**(04): 21–33.

33. Khan-Magomedov, S. (1990). *Vhutemas : Moscou 1920–1930. Vhutemas : Moscou 1920–1930* (üditions du regard ed.): Larroche, Hélène.

34. Rainer, W. (2000). Teaching at the Bauhaus. Teaching at the Bauhaus: Hatje Cantz.

35. Schön, D., *The reflective practitioner*, Basic books, 1983.

36. Talbott, K., Divergent Thinking in the Construction of Architectural Models, *IJAC*, 2004, **2**(2): 264–286.

Ivanka Iordanova
University of Montreal
School of Architecture,
CAD Research Group (GRCAO)
C.P. 6128, succ. Centre-ville Montreal,
Quebec, H3C 3J7
CANADA

ivanka.iordanova@umontreal.ca