

2 Machine Learning and Computational Design

3

4 **Silvio Carta**

5 Proposal for <https://ubiquity.acm.org/index.cfm>

6

7 The use of computers in design is substantially different today from what it was only 30 years ago.

8 And light-years ahead of how things were designed before computers entered the scene only about

9 60 years ago. This article discusses the use of computers, more specifically computational design, as a

10 useful tool for designers (computational design here refers to the application of computational tools
11 to design practice).

12 Design practice often involves long hours devoted to repetitive tasks such as research, testing, and

13 drawing many options in order to work out the best solution for a given problem. This is particularly

14 the case for architecture and construction. For example, if designers working on a new residential

15 building want to find out the optimal slant for each façade panel to maximise solar gain (amount of

16 sunlight entering through each window), they will need to test several strategies, evaluate them,

17 create models, and simulate results in order to compare the efficiency of each option. Once the

18 designers have found the right strategy, they will still need to revisit each individual panel to evaluate

19 the best angle for performing the task. In this way, a single design job could take weeks of testing,

20 adjustments, meetings with consultants, and could easily lead to frustration with the complexity of

21 the entire process.

22 For centuries designers accepted this repetitive and often vexating process, largely because they had

23 no choice. It was either this or not design anything of any interest. For example, Renaissance

24 architects created multiple physical models/maquettes to convince their clients of the aesthetic

25 qualities of their projects. It is therefore not surprising that the automation and optimisation of tasks

26 appeals to designers and others involved in the design process. This is particularly evident in the case

27 of computational design, whereby computers and software are used as a fundamental part of the

28 process.

29

30

31

32 Evolution of computational tools

33 During the 1990s and 2000s, designers started to recognise the benefits of using computers to
34 simplify laborious or complex tasks, to save time and resources, and to acquire a higher level of
35 precision and control over the design process. Notably, architecture firm Gehry and Partners made
36 early use of the parametric software Catia to assist the design and fabrication of the Guggenheim
37 Museum in Bilbao. Around the same time, in 1993, Jon Hirschtick developed Solidworks a CAD
38 (Computer-Aided Design) software that is now used by millions of designers and engineers in product
39 design. The use of CAD, whereby designers use software to replicate hand-drafting more efficiently
40 and accurately, quickly became popular.

41

42 After this initial “digital phase”, which focussed largely on the replication of human tasks by
43 computers, a new way of using computers for design emerged. Recently, a new generation of
44 designers has started including the use of algorithms and computational logic in their work. This
45 approach necessitates a much greater understanding of how computers work and involves the use of
46 computational thinking as a fundamental part of the design process. This new digital age in design
47 includes an awareness of algorithmic logic, datasets and statistic models. In this respect, as design
48 becomes increasingly data-driven, designers find themselves learning more about this data and
49 developing more effective ways of handling them. The technique of form-finding is a clear example of
50 such an approach, where the shape of a building is not created by the designer but by a combination
51 of algorithms. The designers develop a series of tasks for the computer to perform, they set certain
52 conditions, and then they use computers to run a series of simulations/tests that will eventually
53 return the desired shape. Such approaches have been applied to many fields in design, including
54 jewellery, product and furniture design (Philippe Morel’s algorithmic chair is a good example of this),
55 fashion (e.g. 3D printed garments), graphic design (using software like Casey Reas and Ben Fry’s
56 Processing), as well as architecture and construction.

57

58 In general terms, design practice is a vast and complex discipline and it would not be possible to
59 automate or optimise all aspects of it. For example, aspects pertaining to intuition, synthesis and
60 creativity within the design process are hardwired into human nature and cannot be easily replicated
61 by algorithms. However, areas that involve the use of data could be processed by computers and
62 automated in order to augment the design practice. In this sense, computational design should not be
63 considered a substitute for design in general, where automation completely takes over the creative
64 process, it should rather be considered as an additional tool for designers that can, and indeed

65 should, be used to simplify, improve and extend their work. Through computation, designers can
66 perform quicker, more accurate and more comprehensive tasks to test concepts and ideas.

67

68 Machine Learning (ML)

69 One of the most popular and increasingly used computational approaches to design is machine
70 learning (ML). This is where designers and data scientists work together to generate workflows, a
71 combined series of different steps in a process that result in optimised shapes, spatial configurations,
72 and more performant objects. It is not difficult to imagine how the designers frustrated with the angle
73 calculations of multiple facade panels may welcome ML as a very useful tool.

74

75 There are three main types of applications where ML is proving particularly beneficial within design
76 processes, specifically in the Architecture, Engineering and Construction (AEC) industry.

77

78 The first of these are **analytical tools**, where designers use ML techniques to simulate and monitor
79 possible design scenarios. This includes the analysis of existing buildings and public spaces, as well as
80 hypothetical studies, where different factors are tested and building performances evaluated. A
81 recent example of this approach is the MIT SenseAble City Lab's AI Station project which analysed Wifi
82 signals to understand how passengers move through two stations in Paris¹. They used a multi-layered
83 analytical process called Deep Convolutional Neural Network (DCNN) to evaluate indoor legibility in
84 the Gare de Lyon and Gare St. Lazare. Indoor legibility is the extent to which a space is organised in a
85 clear and coherent pattern and can be recognised by users. Researchers in this project used
86 photographic images as an input in order to observe people's behaviours and space utilisation as well
87 as visual portions of the spaces.

88 The second type are **design tools** that have been developed to support designers in their projects and
89 research, mainly running on open platforms. These include Dynamo (an open-source graphical
90 programming tool), Autodesk Revit (one of the main pieces of building information modelling
91 software used widely by architects, mechanical engineers and contractors) and McNeel Rhinoceros'
92 Grasshopper (a visual programming language and 3D modelling software). This group includes
93 applications like Dodo², Owl³ or Lunchbox⁴, where traditional parametric 3D modelling programs can

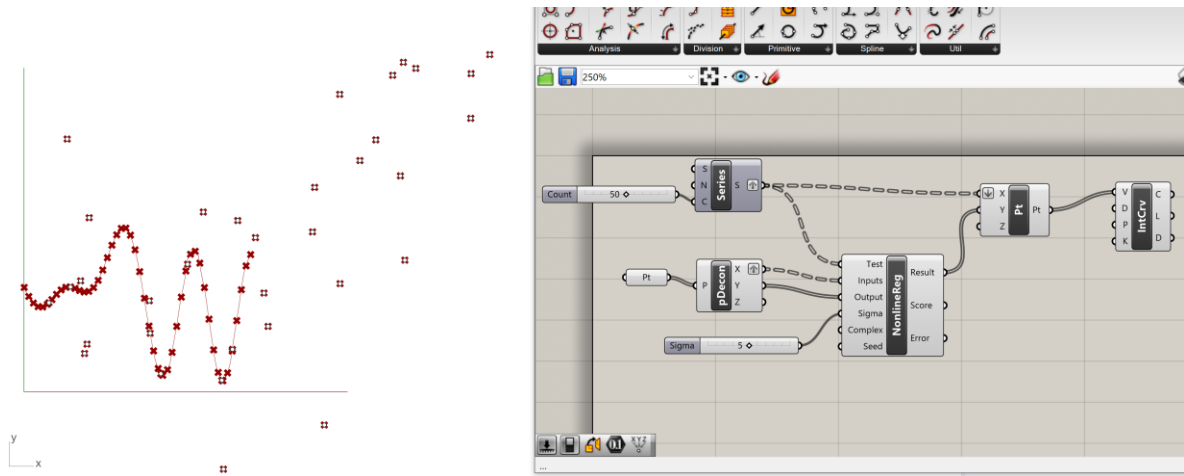
¹ The project is explained in details here: http://senseable.mit.edu/ai-station/app_wifi/

² <https://www.food4rhino.com/app/dodo>

³ <https://www.food4rhino.com/app/owl>

⁴ <https://www.food4rhino.com/app/lunchbox>

94 be augmented by libraries that add machine learning capabilities (e.g. Artificial Neural Network,
95 nonlinear regression, K-Means clustering etc.) to be used in conjunction with spatial data modelling.



96

97 Figure 1. ©Silvio Carta. Example of Rhino/Grasshopper interface (Lunchbox library). The example shows a non-linear regression applied to a
98 random series of points. The algorithm predicts the Y value based on the X value from a training set.

99

100 The third group includes **management and information tools** and can be considered as an extended
101 version of more traditional Building Information Modelling (BIM) systems. These tools are generally
102 referred as part of “City Information Systems” (CIM) and are characterised by a wider application of
103 ML to urban policy and management. New ML-led approaches are being developed across the private
104 and public sectors to combine existing urban information (property, location data, ordinance surveys.)
105 with information generated by the actors involved in the planning process. A particularly successful
106 example is PlanTech⁵, an initiative developed and supported by Connected Places Catapult’s digital
107 planning group. The aim of this project is to foster new ways of managing the public digital
108 infrastructure of planning through increasingly more interconnected databases being used by the
109 different actors involved in the planning process, and more automated and optimised services for
110 final users.

111

112 ML and optimisation for design: A Case Study

113 One particular example may offer more clarity on how machine learning approaches are being used
114 within the Architecture, Engineering and Construction (AEC) industry. In a special issue of the
115 International Journal of Architectural Computing (IJAC), dedicated to the topic of “Intelligent and
116 Informed”, Tarabishy and colleagues presented a ML-based model for effectively computing the

⁵ <https://www.plantechweek.com/>

117 spatial and visual connectivity values in a given space. ~~These are important metrics for developing~~
118 ~~interior layouts but calculating them in real time can be difficult.~~

119 Designers at the global architectural firm Foster + Partners have been working on finding new ways to
120 analyse the spatial configuration of complex building projects (for example large office blocks) at the
121 concept stage of the design process. Spatial configuration is considered in terms of traverse-ability (a
122 measure of how easily users can walk through a space), proximity (the distance between various
123 elements within a space) and visual connectivity (how easily users can see various key parts of the
124 office). The original paper by Tarabishy et al (2019), upon which the following account is based, can
125 be found at this [link](#) and offers a very good example of how contemporary designers and data
126 scientists can work together to optimise their design outcomes.

127 Traditionally, Dijkstra's algorithm has been used to calculate spatial connectivity and visibility graph
128 analysis (VGA) to calculate visual connectivity. Dijkstra's algorithm calculates the shortest paths
129 between nodes in a graph, which may represent for example, various elements in the office, and VGA
130 analyses what people can see in a given space. The problem with these techniques, especially when
131 applied to architecture, is that their computation can be quite heavy and time-consuming making it
132 difficult to provide a real-time response where it is most needed, for example in the sketch/concept
133 design phase. In their study, Tarabishy and colleagues explore the use of ML-based techniques to
134 generate surrogate models which substitute/augment these computationally heavy simulations using
135 deep neural networks. These approaches achieve a significant reduction of the computation time
136 along with an optimisation of the resources required (Tarabishy et al. 2019:54).

137 In their article, Tarabishy and colleagues (2019) begin by explaining how the spatial and visual
138 connectivity for a given floor plan can be calculated using VGA and Dijkstra's algorithm, a lengthy
139 process requiring significant computational resources including hours/weeks of calculations
140 depending on the complexity of the floor plan and the availability of resources.

141 In order to prepare the spatial configuration of a building for simulation and analysis in this way, floor
142 plans need to be reduced to a spatial grid (and parametrised) that includes the key features of the
143 building such as walls, doors, passages, furniture etc. In this particular study each cell is 0.3 metres
144 and is represented as a graph node for the purposes of analysis. Adjacency is calculated as immediate
145 connection with neighbouring cells for the spatial connectivity (excluding unavailable cells like those
146 of walls etc.), and with the rule of "*two nodes are connected to each other if you can draw a line*
147 *without crossing an obstacle for visual connectivity*" (Tarabishy et al. 2019:55). Tarabishy and
148 colleagues used Dijkstra's algorithm to calculate the shortest path within the graph (i.e. traversing the
149 graph), whilst the values of the connectivities were calculated using an isovist graph model.

150 Recognising the computational intensity of these simulations, Tarabishy and colleagues set about
151 trying to improve the calculation of spatial and visual connectivity by using machine learning. They
152 considered this task in terms of a supervised learning problem, using floor plans as images, and
153 approaching the problem in terms of image processing (rather than semantically, as before). This can
154 be thought of as a mapping exercise between an image of a floor plan (used as an input, with key
155 spatial features such as walls and furniture) and an image of an analysed plan (used as an output). The
156 output image has some of its pixels unchanged, namely those representing walls and furniture, and
157 others with new values assigned according to the analysis. These values are represented using colour
158 gradients.

159 Having re-expressed the problem in terms of image processing, Tarabishy and colleagues employed
160 Convolutional Neural Networks (CNNs) to optimise the analysis of spatial and visual connectivity. This
161 method is supported by recent experiments indicating that these CNNs-based algorithms can perform
162 better than others in object detection performance and image classification. In order to be useful for
163 the ML experiment, Tarabishy and colleagues needed to prepare a set of training data in a suitable
164 format. This data consisted of a large number of different floor plans in raster format and with
165 enough resolution to be effectively processed without unnecessary noise. The researchers were then
166 able to carry out a synthetic data generation using an automated system through a CAD framework
167 (Rhinceros and Grasshopper). This parametric model allowed them to generate 6,000 bidimensional
168 plans with a variety of spatial configurations (walls and furniture arrangement) to be used for initial
169 testing (Tarabishy et al. 2019:57). The generated images had a resolution of 100 x 100 pixels (each
170 pixel representing 1 metre of physical office space) and this was considered a good compromise
171 between the indication of key elements in the plan (expressed in binary terms, with black pixels
172 representing walls and un-traversable elements and white pixel for walkable spaces in the grid) and a
173 reasonable analysis time (which grows exponentially with the resolution).

174 In order to improve the analysis of these plans with regards to boundaries and the position of users,
175 Tarabishy and colleagues introduced a signed distance function (SDF) in all generated plans. This
176 function is used to determine the distance of any point x from the closest other fixed point in a set Ω
177 (indicating the office boundary walls). There is evidence to support that the inclusion of the SDF along
178 with a binary system of spatial representation and in conjunction with CNNs can improve the
179 computability of a model for real-time analyses.

180 Once the dataset is ready for inputting there are a number of parameters that need to be considered
181 and tested before starting the actual training. In this project, the learning rate and the choice of the
182 algorithm were amongst the most important of these. Tarabishy and colleagues tested a number of

183 approaches from the U-Net model, a type of CNN, approaches from the U-Net model (a type of CNN
184 based on fully convolutional network FCN that has been developed for biomedical image
185 segmentation and that, compared with the original FCN, outputs more precise segmentations with a
186 smaller number of training images, including stochastic gradient descent (SGD), Adam, RMSProp and
187 Adadelta. Among all these optimisers, SGD and Adadelta performed better, with a rapid convergence
188 (correctly mapping black pixels in input with black pixels in output). Eventually they selected Adadelta
189 to run the experiment on the basis that it presented the fastest convergence rate ie. correctly
190 mapping black pixels in input with black pixels in output than the other options (Tarabishy et al.
191 2019:59).

192 Machines learn by means of a loss function, a method of evaluating how well the algorithm models
193 the given data. Tarabishy and colleagues opted to use a combination of the mean squared error (MSE)
194 and gradient difference loss (GDL) (to introduce a weighted sum) to define the loss function in this
195 case. The latter approach is used in ML, as well as in neural networks, to estimate the performance of
196 a certain model in the optimisation process.

197 In order to obtain the intended level of accuracy , the researchers introduced a generative adversarial
198 networks (GAN) approach based upon two models competing with each other to complete a given
199 task; one generating images that are accurate enough to convince the other model, and a
200 discriminator assessing the outputted images. GAN then converts the loss function into a parameter
201 that can be used to train the model. According to Tarabishy and colleagues (2019:60), *“This*
202 *architecture avoids hand-engineering of the loss function and incentivizes the network to produce*
203 *images which could be undistinguishable from reality.”*

204 The researchers implemented GAN, and carried out this training, with the Pix2Pix architecture. This is
205 where a network maps input to output images, whilst at the same time learning a loss function to
206 train the mapping that allowed them to translate an input image to a specific output (instead of a
207 random image), to become a conditional generative adversarial network (cGAN) (Tarabishy et al.
208 2019:60).

209 Tarabishy and colleagues (2019:61) found that ,[by using] *“the Pix2Pix architecture, inference*
210 *(predicting the output given an input image) for one image is computed in 0.08 s and for the U-Net in*
211 *0.032 s for each of the analyses, compared to 15 s for running the actual spatial connectivity analysis*
212 *and 128 s for running the visual connectivity.”* The results clearly demonstrate how deep learning
213 surrogate models (and more specifically convolutional neural networks) can significantly reduce the
214 calculation time for an analysis of spatial configuration (0.032 seconds versus 15 and 128 seconds of
215 the methods based on graphs and using VGA and the Dijkstra’s algorithm).

216

217 Computational design today and tomorrow

218 This study is fairly representative of the work that progressively hybrid profiles of architects, planners
219 and data scientists are conducting under the umbrella of computational design. Increasingly, global
220 architecture and urban design firms are establishing in-house research clusters to carry out advanced
221 research in data analysis and visualisation, building optimisation, simulation and building
222 performance. Zaha Hadid's Code, KPFui Urban Interface and Foster + Partners' Applied Research and
223 Development group are all well known examples of this trend. Research like the study conducted by
224 Tarabishy and colleagues is increasingly relevant both for designers and, more importantly, for
225 everyone involved in the planning, management and use of cities and public spaces. Research into
226 optimisation within design is helping to produce quicker and more accurate simulations, tests and
227 prototypes in projects where each decision in the design phase corresponds to a large number of
228 actions, costs, months and years of work, and resources. Simulations and analyses of buildings and
229 cities are becoming increasingly more precise, leaving a smaller margin for error and human mistakes.
230 Automation and optimisation of processes in design yield better outcomes (that are more
231 performant, functional and appreciated by users).

232 There are many aspects of design that are not computational and still rely on human perception,
233 taste, preference and intuition. However, the systems for the computational aspects have come a
234 long way and will allow for much more complex systems in the future.

235 A number of challenges still exist for the years ahead. Such approaches are still generally sporadic and
236 characteristic of only a small number of cutting-edge research groups within traditional design firms
237 and universities. In other words, research in optimisation, classification, sorting and in machine
238 learning more generally are only possible today within those practices and institutions that can allow
239 investment, in terms of time and resources, into computational research. This tends to occur on a
240 centralised level (research centres, universities and large design consultancy companies), and is much
241 more difficult (and rare) for small-medium design practices, start-ups and individuals to engage in. If
242 this line of research is considered to be vital for the progress of design, we are still quite far from
243 reaching a critical mass whereby computational design becomes a collective effort, shared by the
244 entire global design community instead of being promoted by a few small groups of excellence.

245 There may be a silver lining though. As technology progresses at a fast pace, the Architecture,
246 Engineering and Construction (AEC) industry is constantly pressured to embrace new ways for
247 processes to be automated and optimised, projects to be planned and controlled with higher

248 accuracy, and new data to be produced around each design process (from the exact quantity of
249 certain building materials present in a construction site to metrics to monitor the user's experience in
250 cities). The next 5-10 years will be characterised by an increase in attention to computational design,
251 optimisation and ML techniques to support design. Cities are likely to be increasingly governed by
252 intelligent systems where ML will be a fundamental component, and young designers currently
253 enrolled in a growing number of new university programmes that include computational design in
254 their curricula, will be able to contribute more significantly to urban projects. They will consider ML as
255 one of many options in their design toolbox, therefore normalising and extending the use of ML
256 within the design process. This would allow the extension of computational design and ML to a larger
257 platform, whereby the number of designers engaging in the use of (and proportionally in the research
258 associated with) machine learning reach a critical mass extending to small groups and individuals as
259 well.

260 Acknowledgements

261 The central part of this article is dedicated to the description of a project carried out by Sherif
262 Tarabishy, Stamatios Psarras, Marcin Kosicki and Martha Tsigkari and published in 2019 in the
263 International Journal of Architectural Computing (IJAC) doi:
264 <https://doi.org/10.1177/1478077119894483>. I would like to thank Sherif Tarabishy at the Applied
265 Research and Development group, Foster + Partners, London, UK for his support with this article.
266 I would also like to thank the Editors at Ubiquity for their useful comments and suggestions, in
267 particular Alessio Malizia who supported me with fruitful discussions and good advices. Finally, I
268 would like to thank Anouska Plaut for her help with the overall structure of the article.

269

270 References

271 Tarabishy, Sherif, Stamatios Psarras, Marcin Kosicki, and Martha Tsigkari. "Deep learning surrogate
272 models for spatial and visual connectivity." *International Journal of Architectural Computing* (2019):
273 1478077119894483.