Architecture & Style

A New Frontier for AI in Architecture

Stanislas Chaillou | Harvard Graduate School of Design | June 2nd, 2019

GAN-Generated Apartment Units, with Specific Styles | Source: Author

In this article, we release a part of our thesis, developed at Harvard, and submitted in May 2019. This piece is one building block of a larger body of work, investigating AI's inception in Architecture, its historical background, and its potential for space organization & style.

We build here upon a previous piece, where our emphasis revolved around the strict organization of floor plans and their generation, using Artificial intelligence, and more specifically Generative Adversarial Neural Networks (GANs). As we refine our ability to generate floor plans, we raise the question of the bias intrinsic to our models and offer here to extend our study beyond the simple imperative of organization. **We investigate architectural style learning, by training and tuning** an array of models on specific styles: *Baroque, Row House, Victorian Suburban House, & Manhattan Unit*. Beyond the simple gimmick of each style, our study reveals the deeper meaning of stylistic: more than its mere cultural significance, style carries a fundamental set of functional rules that defines a clear mechanic of space and controls the internal organization of the plan. **In this new article,we will try to evidence the profound impact of architectural style on the composition of floor plans.**

Reminder: AI & Generative Adversarial Neural Networks

While studying AI and its potential integration to the architectural practice, we have built an entire generation methodology, using **Generative Adversarial Neural Networks (GANs)**. This subfield of AI has proven to yield tremendous results when applied to two-dimensional generation of information. As any machine-learning model, GANs learn statistically significant phenomena among data presented to them. Their structure, however, represents a breakthrough: made of two key models, the *Generator* and the *Discriminator*, GANs leverage a feedback loop between both models to refine their ability to generate relevant images. The Discriminator is trained to recognize images from a set of data. Properly trained, this model is able to distinguish between a real example, taken out of the dataset, from a "*fake*" image, foreign to the dataset. The Generator, however, is trained to create images resembling images from the same dataset. **As the Generator creates images, the Discriminator provides it with some feedback about the quality of its output. In response, the Generator adapts to produce even more realistic images.** Through this feedback loop, a GAN progressively builds up its ability to create relevant synthetic images, factoring in phenomena found among observed data.

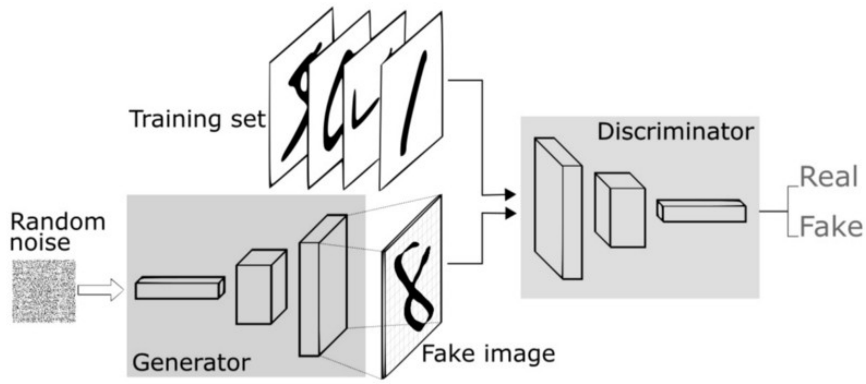


Figure 1: **Generative Adversarial Neural Network's Architecture** | Image Source

We specifically apply this technology to floor plan design, using image representations of plans as data format for both our GAN-models' inputs and outputs. The framework being employed all across our work is Pix2Pix, a standard GAN model, geared towards image-to-image translation.

. . .

I. Organization

The layout in space of elements is a key concern for the architectural discipline. As evidenced *in a previous article*, this exercise is non-trivial one and can be discretized in a sequence of steps. Each step is in fact captured by a thoroughly trained GAN model. The careful study of the organization learned by each model reveals the existence of a deeper bias, know to our discipline as *architectural style*. Unpacking our "generation stack" will help us coin down the different levels of stylistic, and isolate the underpinnings of style's influence.

Pipeline & Generation

At first, we propose a pipeline, where by nesting successive models one after the other we assist the architect in generating a coherent room layout (Model I), and furnishing (Model II), to finally reassemble all apartment units into a tentative floor plan, as shown in Figure 2.

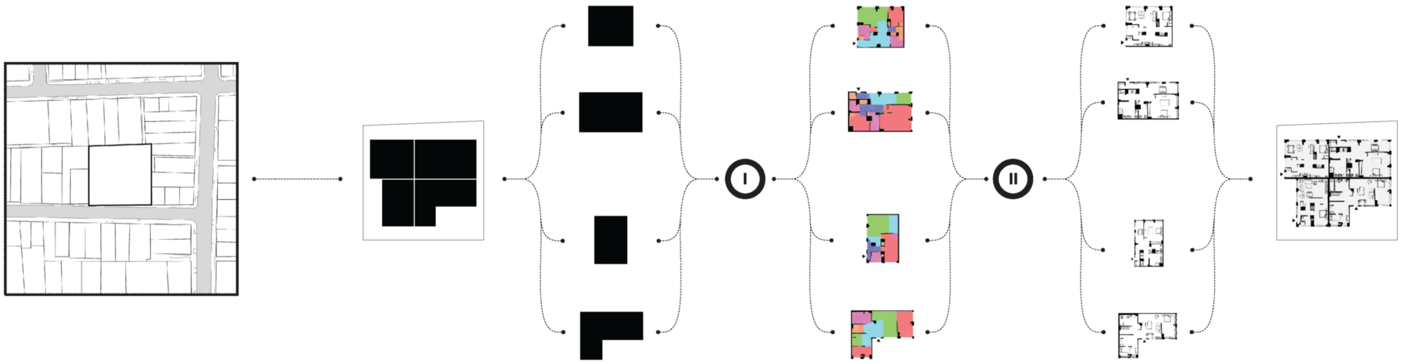


Figure 2: **Generation Pipeline**, Model I & Model II | Source: Author

For each step throughout our pipeline, we provide the user with a simple interface. On the left, he/she can input a set of constraints and boundaries, to generate on the right side the resulting plan. The designer can then iteratively modify the input, on the left, to refine the result, on the right. The animations in Figure 3 showcase this type of interface & process set up for **Model I.**

Figure 3: **Model I** Interface | Source: Author

Bias, or The Emergence of Style

Taking into consideration a batch of generated units, **we start to notice some amount of intrinsic bias to our model**: the internal wall structure is consistently laid out as an orthogonal system of partitions, disregarding the potential orientation of the units' facades,a s shown in Figure 4. At the same time, the program layout is also consistently set up such that "*serving*" spaces -*bathroom, toilets, closets, kitchen*- are packed at the back of the plot, while the odd geometry of the facade gets absorbed by over-dimensioned living rooms and bedrooms.

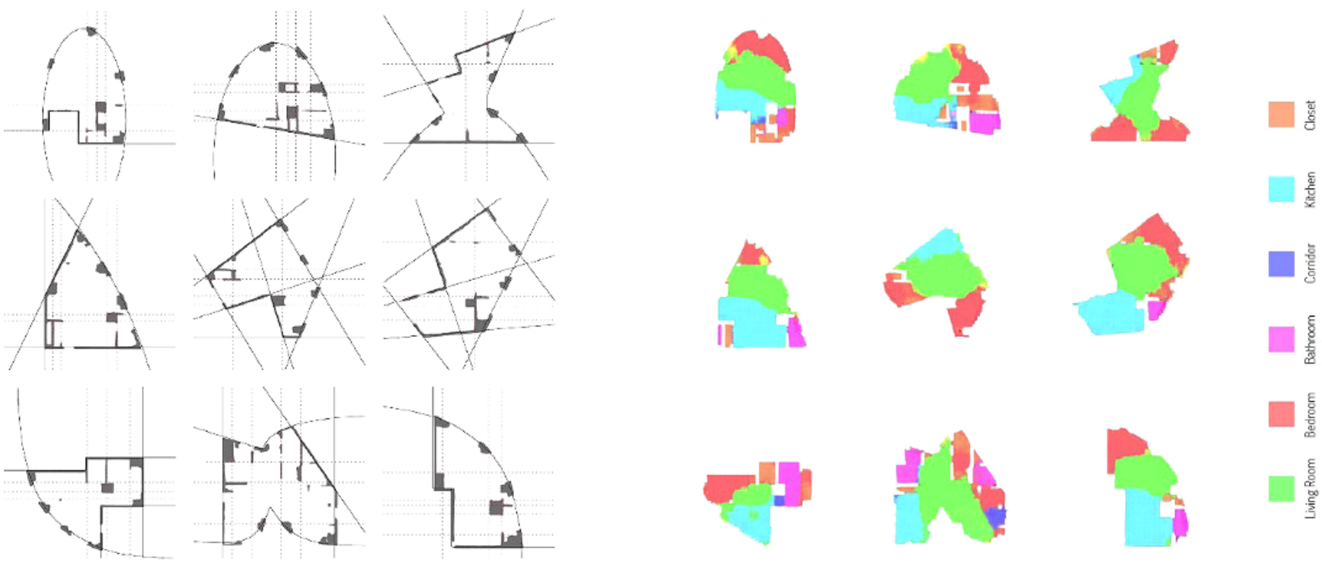


Figure 4: **Plan Wireframe** (left) & **Program Repartition** (right) of Generated Apartment Units | Source: Author

These characteristics can, in fact, be found all across our initial training set. We understand here this reality as the literal translation of a concept central to the architectural discipline: **style**.

II. Style(S)

Instead of preventing this bias, striving to create a *generic* or *objective* plan generator –which is not our concern here- we will rather embrace it and study its presence to eventually use it to our advantage.

To that end, we choose to broaden our investigation and extend it to architectural style learning. We create a new pipeline, enabling the conversion of floor plans, from one style to another, here from *Modern* to *Baroque*.

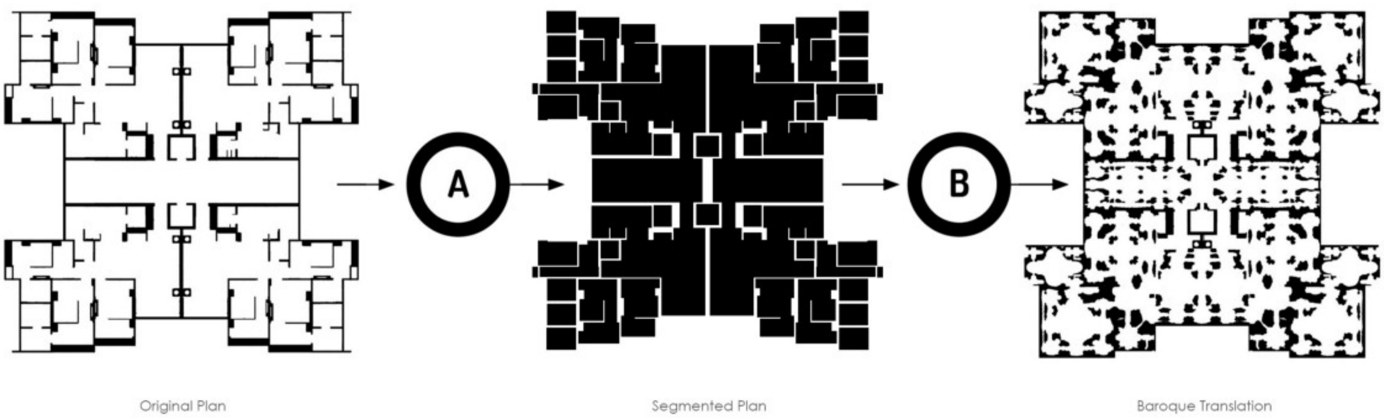


Figure 5: **Modern-to-Baroque** Translation & Subtraction | Source: Author

Figure 5 reveals all the more the deeper meaning of architectural style: in this GIF, we have subtracted the initial wall structure (Modern) from the translated plan (Baroque). The remaining "*poché*" is the addition that a baroque style induces: **it is not simply a new make-over of the existing figure wall, but rather a profound remodeling of internal structures and spatial organizations.**

In fact, we evidence here what Farshid Moussavi coined down as the *Function of Style* in her book. **Each style, beyond its cultural significance, handles space differently and reacts specifically to similar constraints.**

To investigate architectural style learning, we have trained and tuned an array of models on specific styles—*Baroque, Row House, Victorian Suburban House, & Manhattan Unit*—able to emulate each particular architectural style.

Figure 6: **4 Styles** | Source: Author

For each style, we display in Figure 7 to 10 the **initial training set** (left), **some resulting generated apartment units** (center), and a physical model of the same apartment units (right).

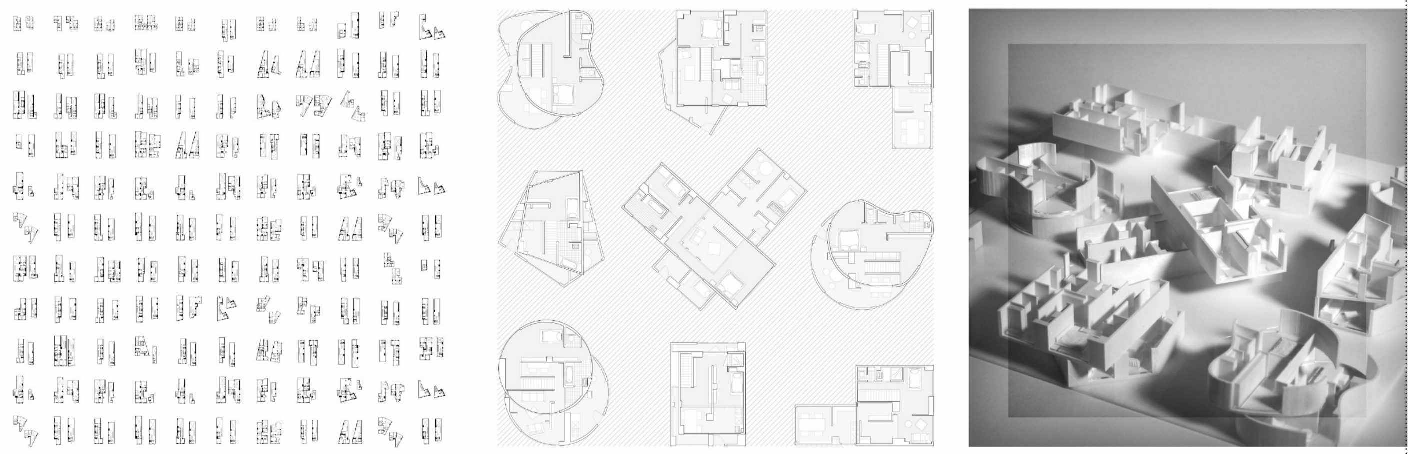
Figure 7: **Baroque** Apartment Units | Source: Author

Manhattan Style Training Set

Generated Units

Generated Units, Physical Model

Figure 8: **Manhattan** Apartment Units | Source: Author



Two-Story Style Training Set

Generated Units

Generated Units, Physical Model

Figure 9: **Row-House** Apartment Units | Source: Author

Victorian House Style Training Set

Generated Units

Generated Units, Physical Model

Figure 10: **Victorian** Apartment Units | Source: Author

Among the generated units we can identify some clear patterns within each style. This “behavior” proper to each model is to us a direct translation of each style’s mechanic. More than applying a simple texture across each apartment unit, each model has captured a set of characteristics and rules.

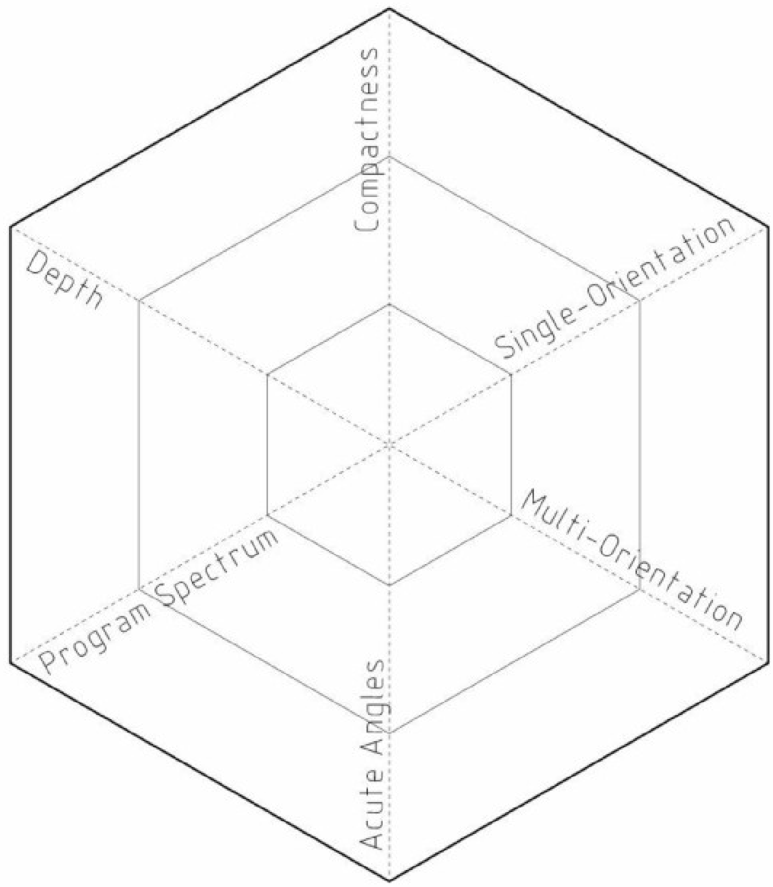


Figure 11: **Characteristics Graph Template** | Source: Author

To move beyond this simple observation, **we offer to map out each style’s abilities.** To each model corresponds a set of strengths & weaknesses, and coining them down will allow us to truly assess the functional reality of each style. In addition, our hope is to expand our understanding of our models’ abilities to allow us later to use each one purposely, given a new set of constraints, and functional requirements.

In clear **we propose a six-axis graph, reflecting a given model's ability to handle six specific types of condition**: *Depth*, *Compactness*, *Single-Orientation* or *Multi-Orientation* (number of facades), *Acute Angle* (sharp geometry of the boundary), *Program Spectrum* (breadth of the program).

After thoroughly testing our four models, we propose the following graphs…

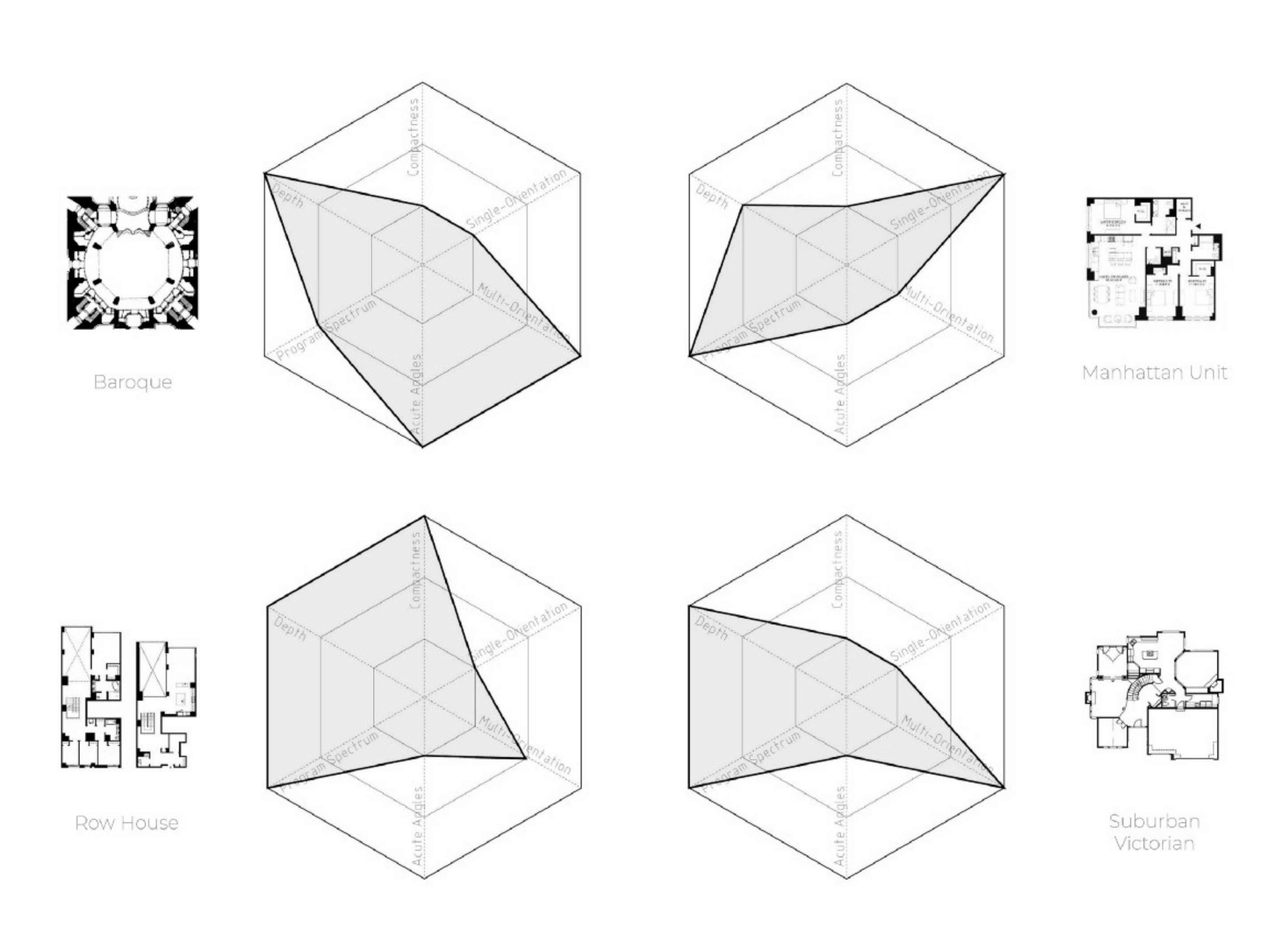


Figure 12: **Styles' Characteristics Graphs** | Source: Author

More specifically, and to explicit Figure 12, **we offer an in-depth explanation of each style's characteristics:**

Baroque Style

- Can handle depth and sharp geometric boundaries. By multiplying niches and alcoves, this style can subdivide the plot, and carve suitable spaces out of the parcel's depth.
- However, this style needs space to unfold its logic & internal structure and therefore does not respond well to compact footprints.

Manhattan Style

- Reacts better to single or double orientations.
- Displays a wide programmatic spectrum.
- Has issues dealing with depth and sharp geometric boundaries.

Row-House Style

- Can make the most out of compact footprints, by spreading functionality between floors and confining each function into small secluded spaces.
- Depth gets handled quite well and gets often filled with a vertical circulation, around which serving spaces are aggregated.

Suburban Victorian Style

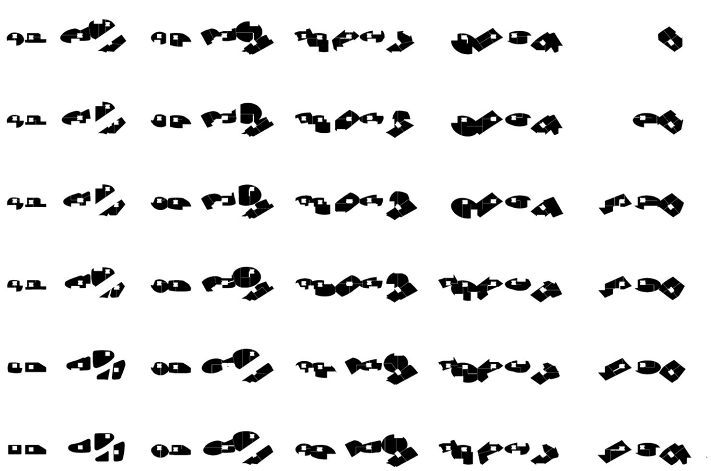
- Handles multiple orientations easily.
- Deals well with depth, by inhabiting the center of the plot with serving spaces.
- Displays a vast spectrum of rooms and a programmatic richness
- However, this style reacts poorly to compact footprints, and cannot truly cope with sharp geometric boundaries.

Figure 13: **Building Massing** (North, East, South & West Elevations) | Source: Author

III. Application

Finally, we have brought all these intuitions together in a final architectural project: a large-scale housing development located in Manhattan's Lower East Side. The complex geometry of the parcel forces a certain amount of complexity on our design. As a result of our massing (Figure 13), we obtain a catalog of 380 one-of-kind apartment units, displayed in Figure 14.

A. A Building



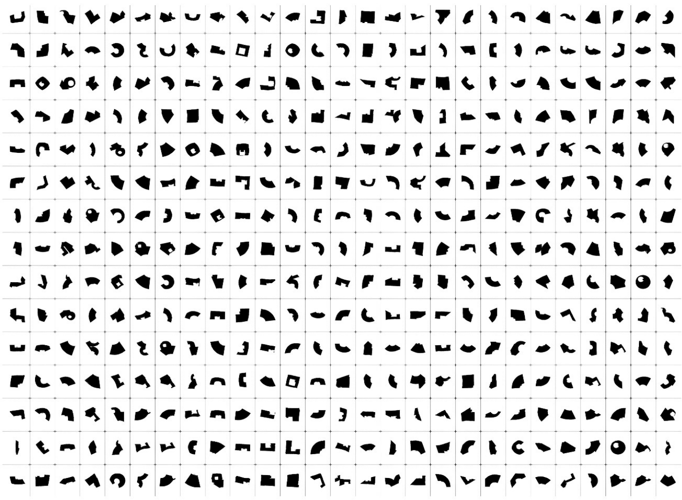


Figure 14: **Sequence Of All Floors** (Left), Catalog of **Floorplates** (Center) and **Units** (Right) | Source: Autho

We first attempt to process an entire floor, each time using a different style. **The results displayed in Figure 15 reveal once again the necessity of using styles cautiously, respectful of the constraints and the specificity of each context.** If certain units are successfully laid out using a certain style, others fail to find a proper internal organization. **By alternating styles across our catalog of apartment units, we hope to find an appropriate answer to each specific spatial condition.**

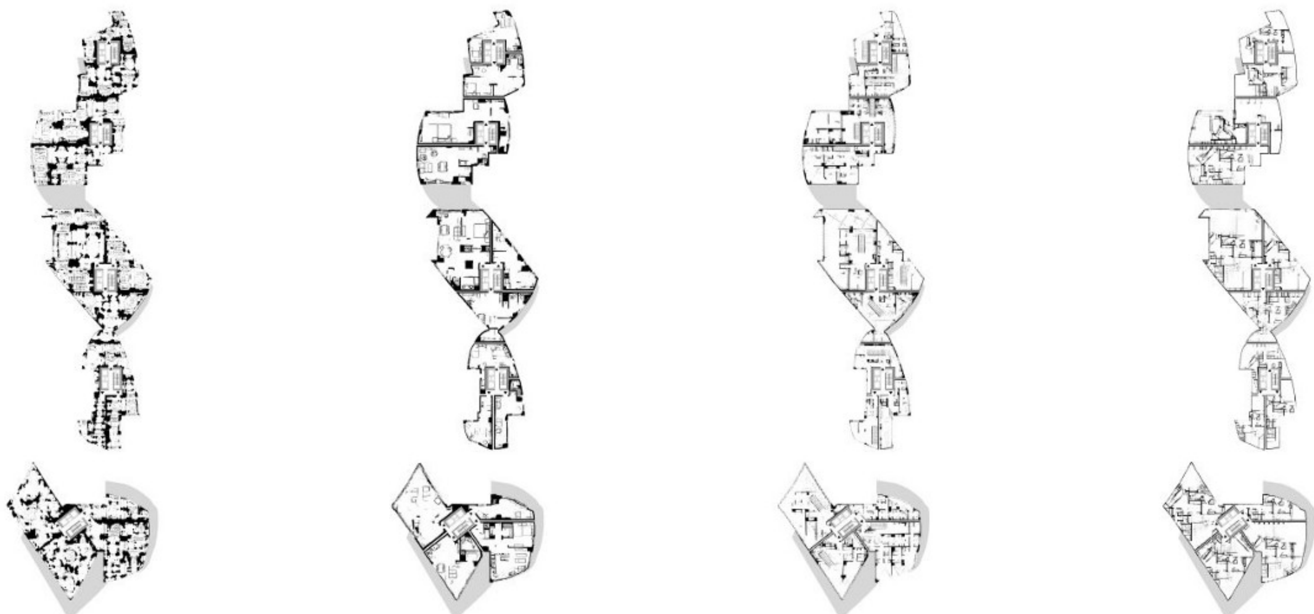


Figure 15: **15th Floor Processed Under Each Style**: Baroque (Far-Left), Manhattan (Center-Left), Row-House (Center-Right), Victorian (Far-Right) | Source: Author

Knowing each GAN-model's strengths and weaknesses, each style's potential and short-comings, we now process every apartment using the most suited model. Each floor is turned into a patchwork of styles. **Our goal becomes then to compose our "mosaic" picking for each tile -each unit- the most reasonable model that will best handle the constraints.** Out of this selection process, we have isolated some resulting options, displayed in Figure 16.

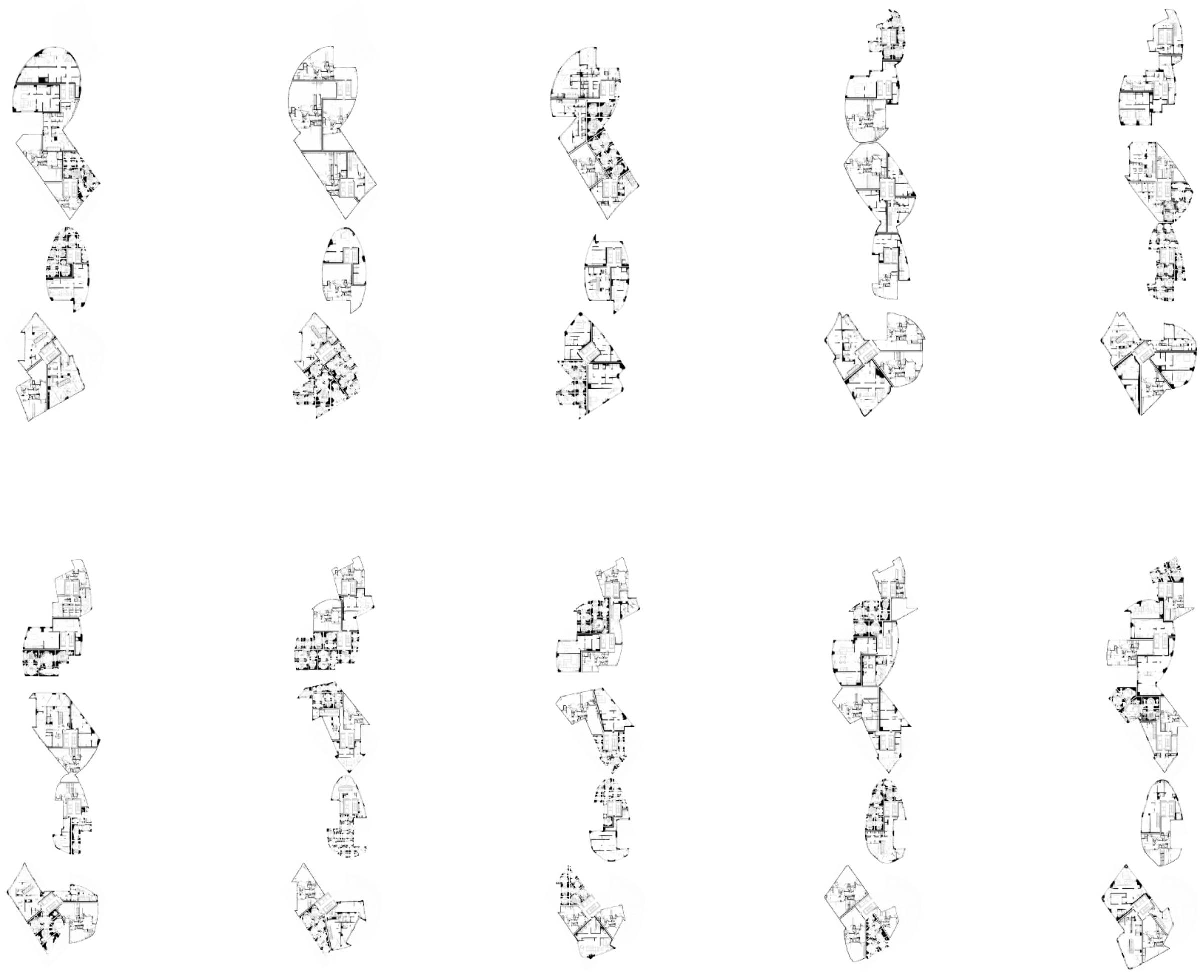


Figure 16: **3 Options: 10th Floor to 20th Floor**, Processed Differently | Source: Author

We then narrow down our exploration, precise the selection of units and styles across all floorplates, to finally assemble our final design. In Figure 17 are displayed three typical floors…

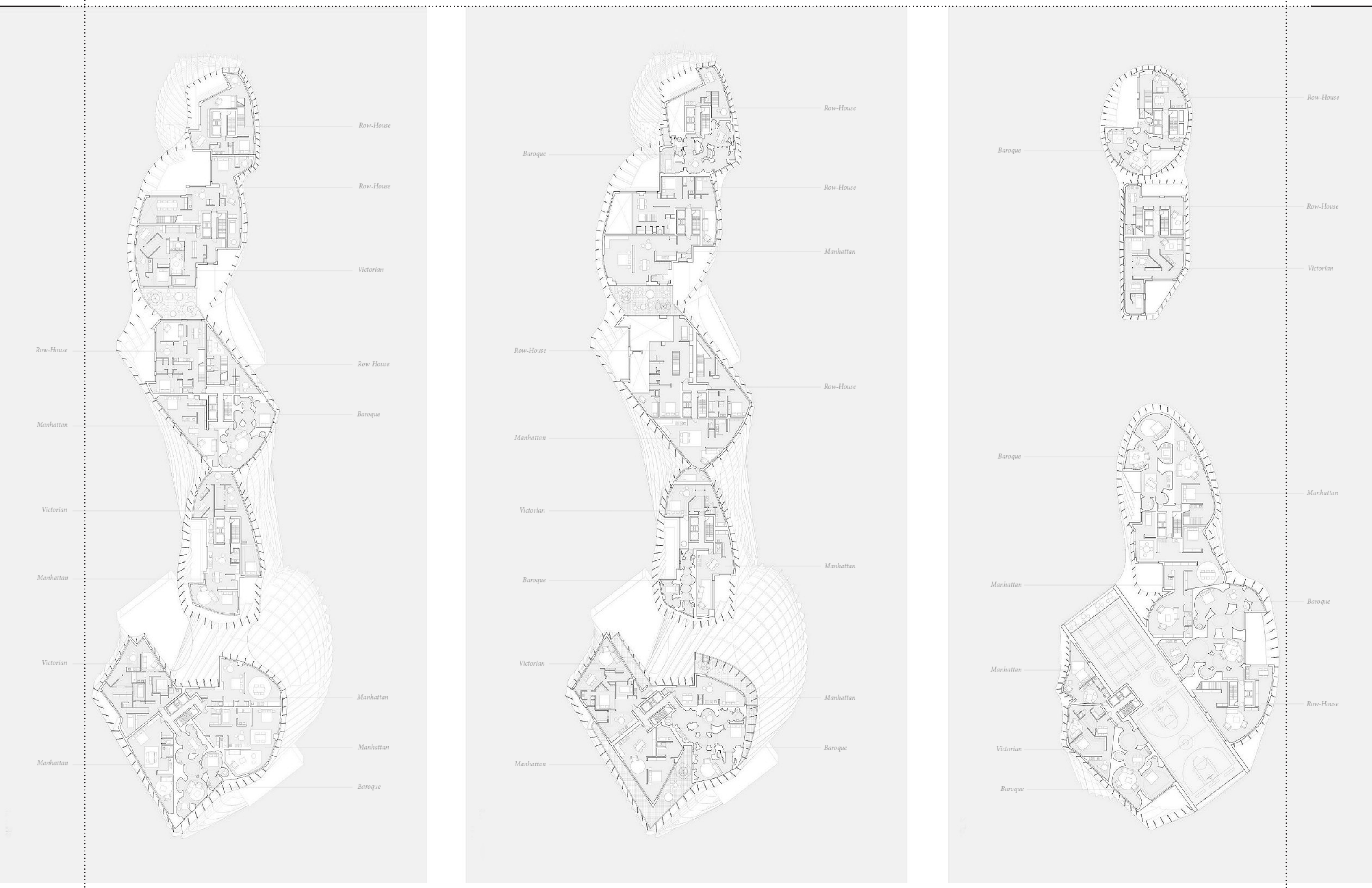


Figure 17: **14th, 15th & 4th Story Floor plans** | Source: Author

We have in fact turned styles into functional tools, able to address specific conditions all across our development.

However, as Figure 17 suggests, **we set aside certain constraints, and make clear assumptions**. We would like to clarify these.

i. The Structure

The structure is left to the structural cores, and the tensions cables running along the facade. **The plan is therefore uninterrupted by vertical loads and allows our algorithms to generate freely each unit's partitioning system.**

__Note__: Looking back to our generation pipeline, a potential improvement would take the position of load-bearing walls & columns as inputs for Model I. In such a way, our pipeline would allow designers to control the building structural system. Alternatively, Model I could be unpacked into two successive models, one for laying out load-bearing elements, and one for adding partitioning walls.

ii. The Strict Imperative of Efficiency

Both the exuberance of each style and the level of freedom given to our models do not address a concern common in our discipline: **space efficiency**. However, our primary concern here is to maximize the expressiveness of each style, to let each model unfold its mechanic to showcase its "personality".

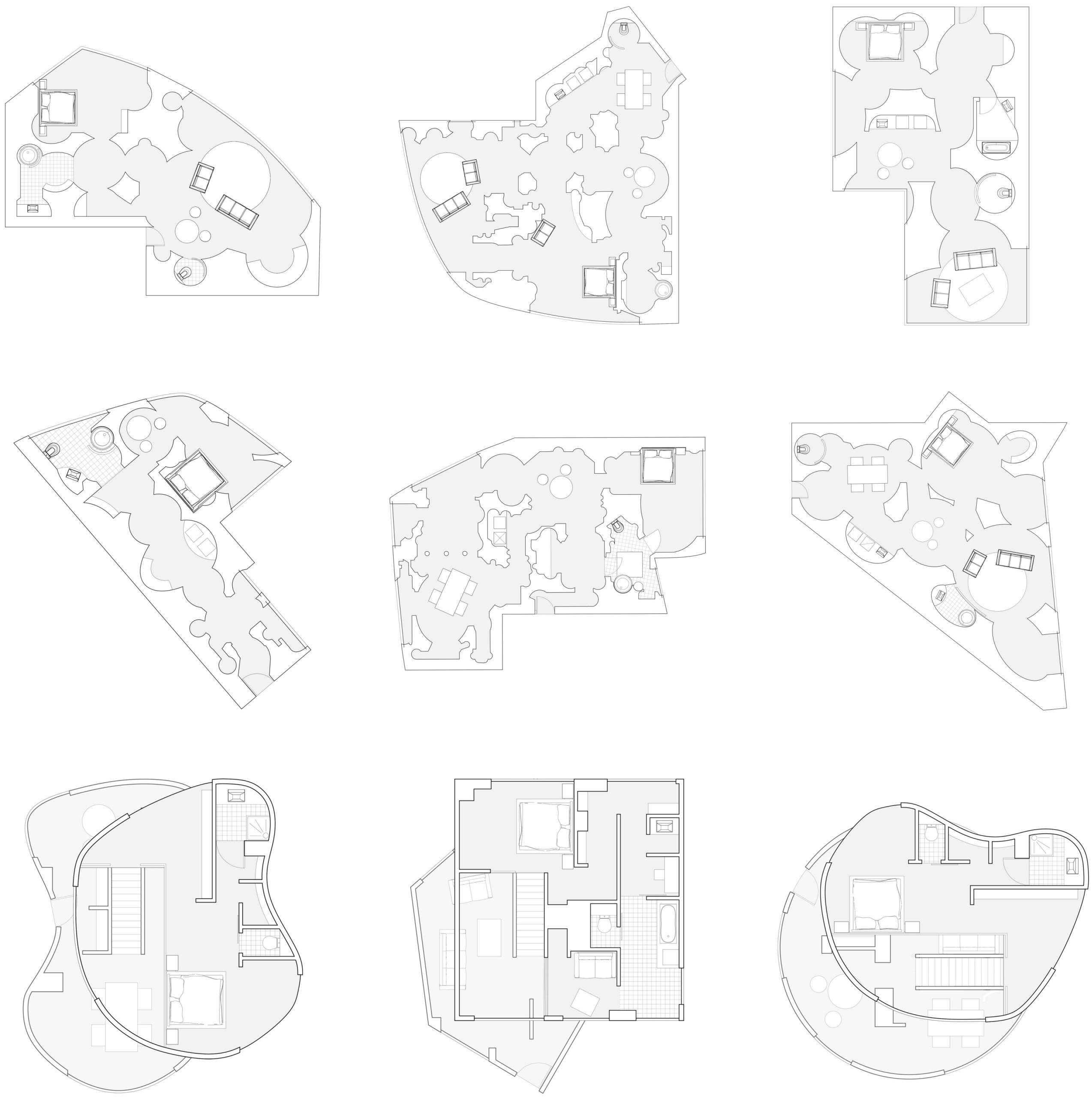
__Note__: To reconcile GANs with efficiency, we posit that their outputs constitute a tremendous initialization for standard optimization techniques. Parametricism's typical pitfall was to set up a too-broad problem space, coupled with a random initialization. Optimizations run on these settings would often end up converging on locally minimal solutions. The intuition behind our GANs brings a whole new quality of initialization, that narrows down significantly the problem space while setting up a good-enough initial solution.

iii. The Massing

The notion of *Massing* refers to the outer-shape of our building. The irrational form of our design is here meant as a trigger for complexity, creating an actual challenge for our models. To a more rationale massing would correspond more tamed & realistic apartment units designs. Our focus here remains to showcase an extreme case and test our models' limits.

B. A Catalog

We turn finally to the previous catalog of units, found across our building. The coherence and richness of the resulting designs are striking. Moreover, the "*intelligence*" or formal flexibility displayed in the generated apartments further evidences the validity of the approach: **GAN-models can indeed encapsulate some amount of architectural expertise & stylistic that can be later used, depending on the set of constraints at play.** The "personality" of each model described in Part II is clearly legible among each subset.



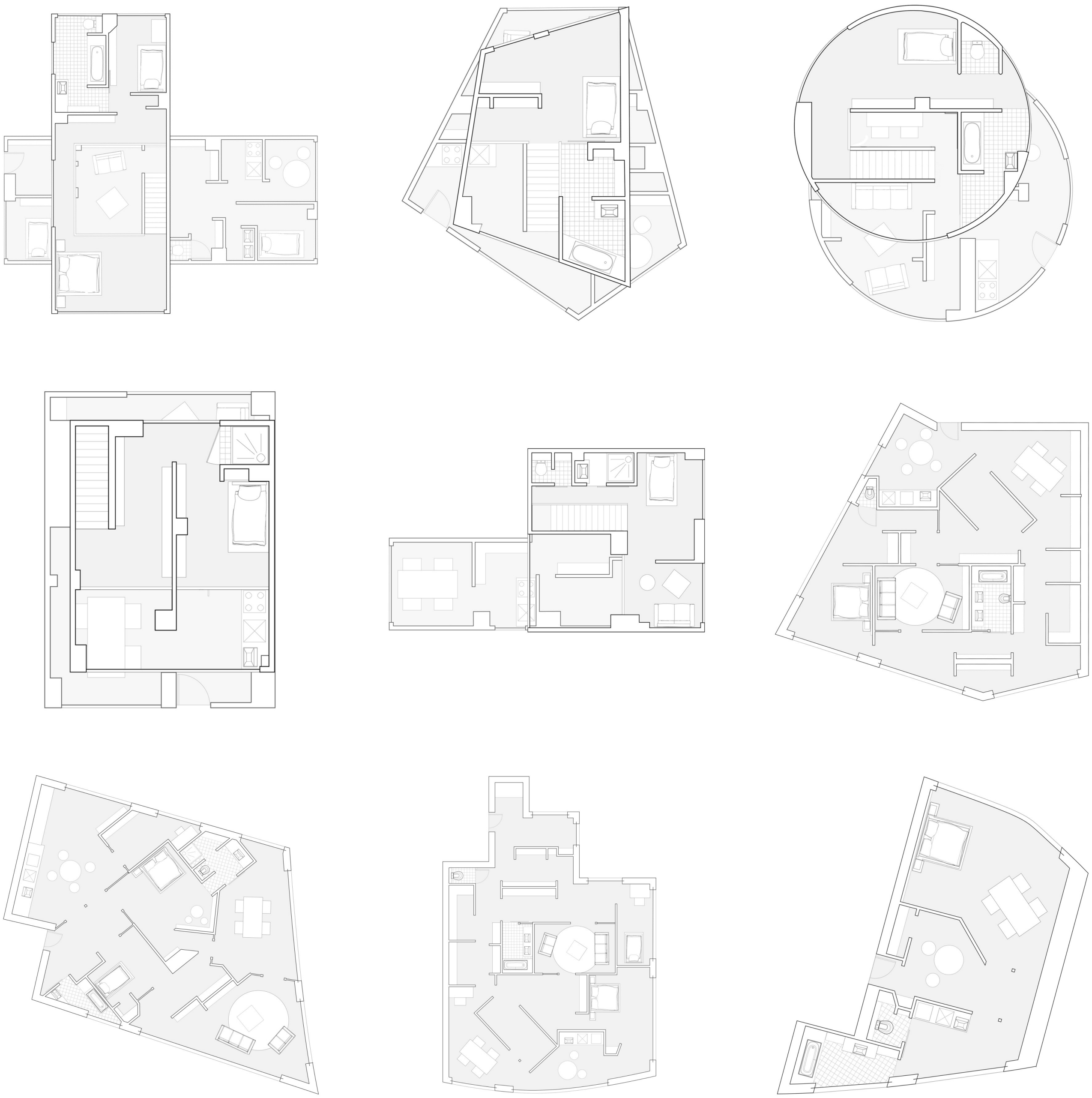


Figure 18: **Generated Apartment Units Catalog** | Source: Author

To conclude this section, **we offer in Figure 19 to 22 a sequence of key shots, taken across our catalog.** To the strict descriptive nature of the plans (*left*), we associate each time an image of the interior (*right*), as a way to reconcile our process with the more experiential nature of Architecture.

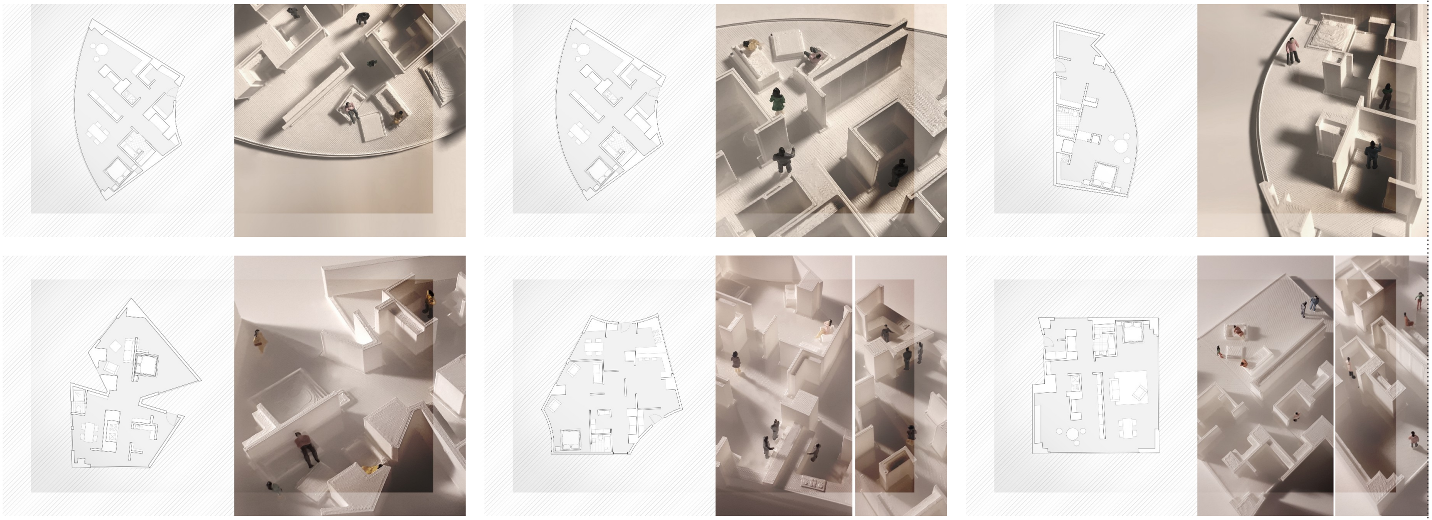


Figure 20: **Manhattan** Apartment Units' Interior | Source: Author

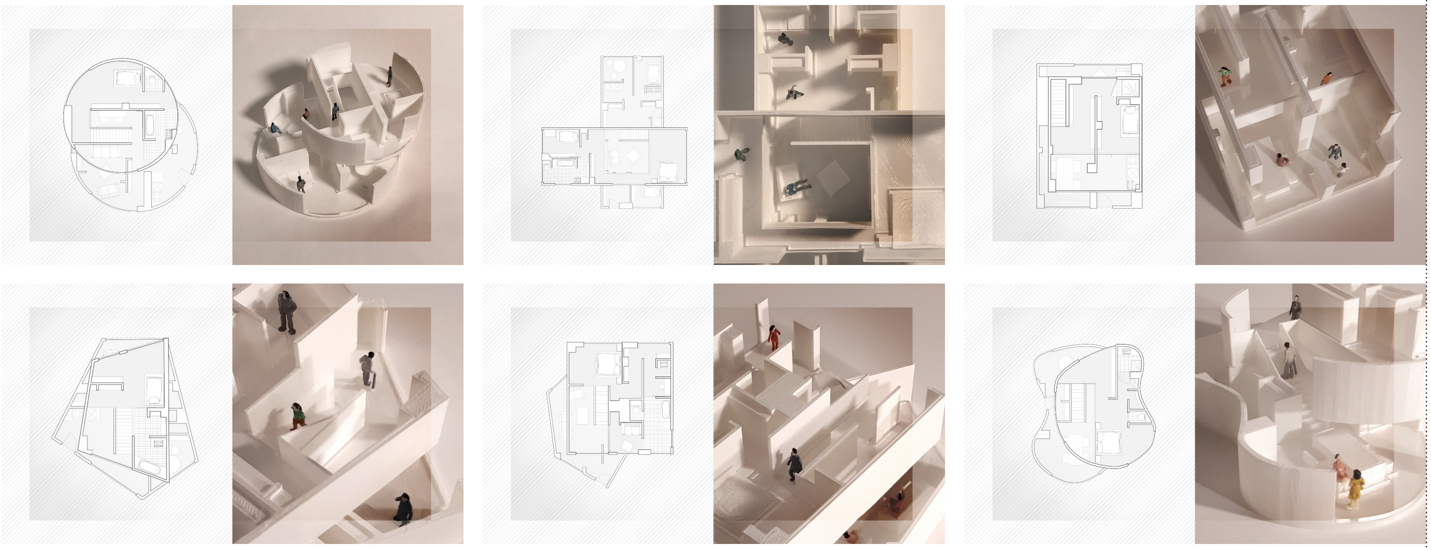


Figure 21: **Row-House** Apartment Units' Interior | Source: Author

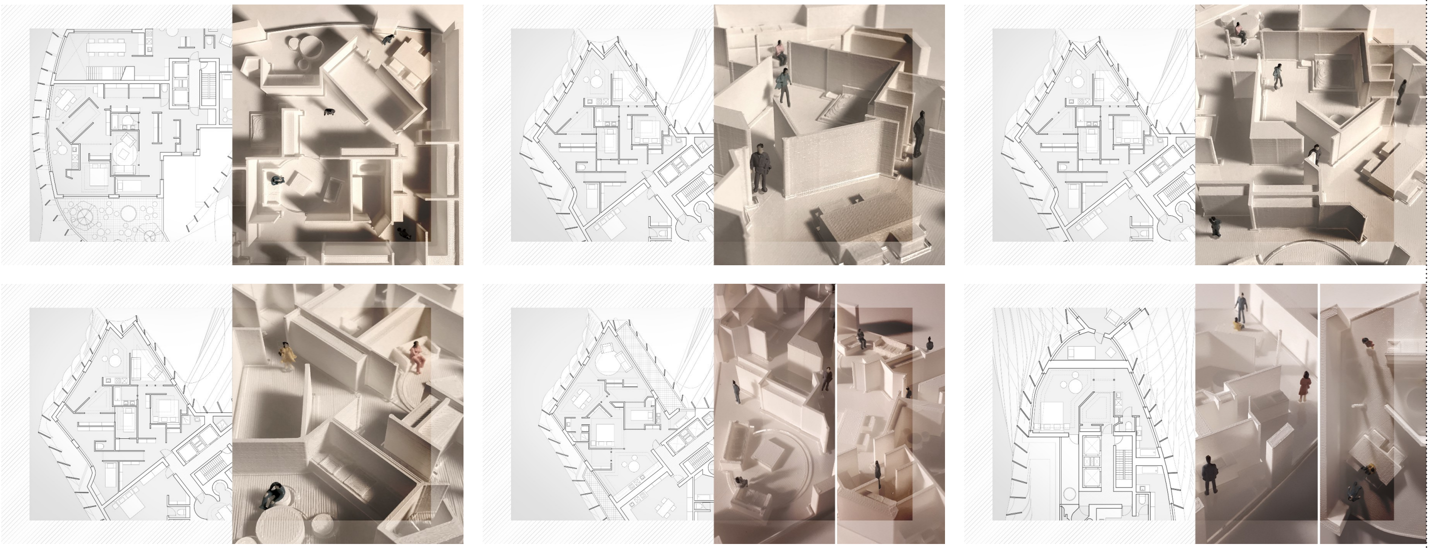


Figure 22: **Victorian** Apartment Units' Interior | Source: Author

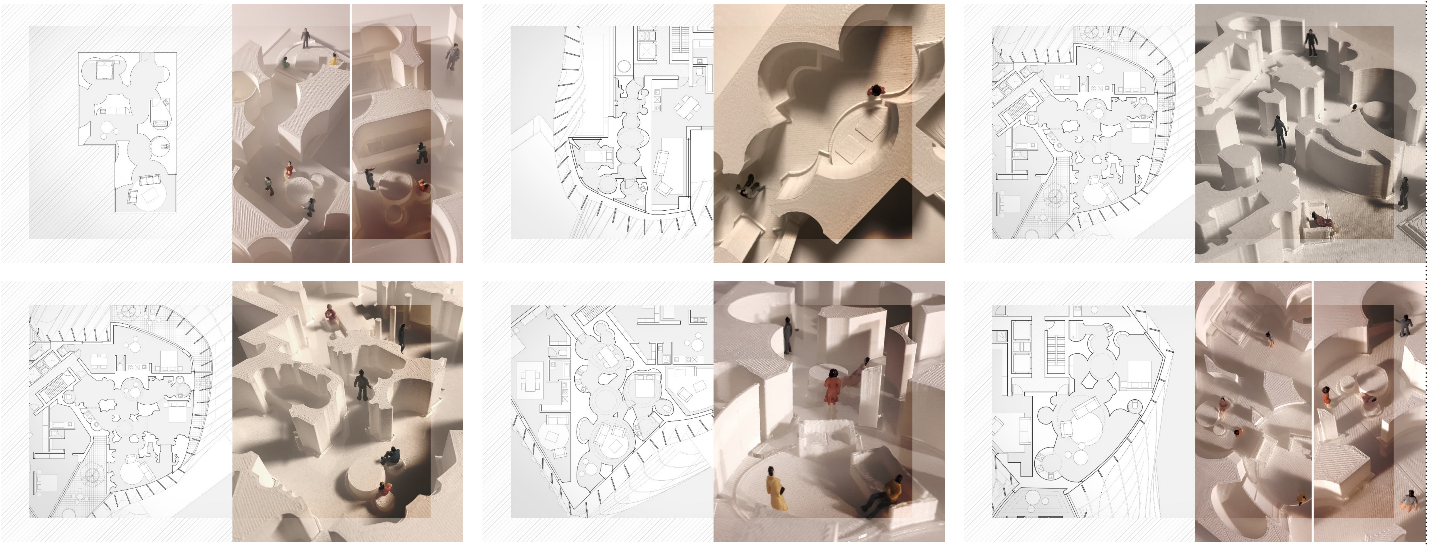


Figure 19: **Baroque** Apartment Units' Interior | Source: Author

IV. Conclusion

If we can think of floor plans first as compositions, before being strictly the product of engineering, then **studying the driving forces of the composition is maybe where AI can offer us some meaningful answers. Following this intuition, we have evidenced in this article that** architectural styles carry at a deeper level an implicit mechanic of space, that significantly impacts any floor plan's composition. In clear, there are spatial consequences to choosing a given style over another.

At a more fundamental level, we can think of styles as being the by-product of architectural history. If there is within each style a deeper set of functional rules, **then studying architectural history could potentially be about understanding the evolution over time of these implicit rules.** Being able to encapsulate each style could allow us to go beyond the study of precedents, and complement it by unpacking the behavior of GAN-models such as the ones trained here. **Their ability to emulate some of the unspoken rules of architecture could allow us to address the "*quality with no name*"**embedded in buildings that Christopher Alexander defines in his book *The Timeless Way of Building.* AI is simply a new way to study it.

Finally, the inherent presence of style within each GAN-model constitutes a final takeaway: far from the promise of an agnostic & objective practice of generative design, it seems that style permeates irrevocably the very essence of any generative process. In clear: Style is not an ancillary, superficial or decorative addendum. Style is at the core of the composition. Recognizing this evidence is a prerequisite to understanding what AI can bring to Architecture. In other words, there will be no agnostic-AI for Architecture, no style-less machine, no objective generative design. On the contrary, each model or algorithm will come with its flavor, its personality, its know-how.

Digital Architecture Beyond Computers, Roberto Botazzi, Bloomsbury

Data-Driven Design & Construction, Randy Deutsch, Wiley

Architectural Intelligence, How Designers and Architects Created the Digital Landscape, Molly Wright Steenson, MIT Press

Architectural Google, *Beyond the Grid—Architecture & Information Technology pp. 226–229*, Ludger Hovestadt, Birkhauser

Algorithmic Complexity: Out of Nowhere, *Complexity, Design Strategy & World View pp. 75–86*, Andrea Gleiniger & Georg Vrachliotis, Birkhauser

Code & Machine, *Code, Between Operation & Narration pp. 41–53*, Andrea Gleiniger & Georg Vrachliotis, Birkhauser

Gropius' Question or On Revealing And Concealing Code in Architecture And Art, *Code, Between Operation & Narration pp. 75–89*, Andrea Gleiniger & Georg Vrachliotis, Birkhauser

Soft Architecture Machines, Nicholas Negroponte, MIT Press

The Architecture Machine, Nicholas Negroponte, MIT Press

Cartogramic Metamorphologies; or Enter the RoweBot, Andrew Witt, Log #36

Grey Boxing, Andrew Witt, Log #43

Suggestive Drawing Among Human and Artificial Intelligences, Nono Martinez, Harvard GSD Thesis, 2016

Enabling Alternative Architectures: Collaborative Frameworks for Participatory Design, Nathan Peters, Harvard GSD Thesis, 2017

Architectural Drawings Recognition and Generation through Machine Learning, Hao Zheng (University of Pennsylvania), Weixin Huang (Tsinghua University), ACADIA 2018

DANIEL: A Deep Architecture for Automatic Analysis and Retrieval of Building Floor Plans, Divya Sharma, Nitin Gupta, Chiranjoy Chattopadhyay, Sameep Mehta, 2017, IBM Research, IIT Jodhpur

Automatic Room Detection and Room Labeling from Architectural Floor Plans, Sheraz Ahmed, Marcus Liwicki, Markus Weber, Andreas Dengel, 2012, University of Kaiserslautern

Automatic Interpretation of Floor Plans Using Spatial Indexing, Hanan Samet, Aya Soffer, 1994, University of Maryland

Parsing Floor Plan Images, Samuel Dodge, Jiu Xu, Bjorn Stenger, 2016, Arizona State University, Rakuten Institute of Technology

Project Discover: An Application of Generative Design for Architectural Space Planning, Danil Nagy, Damon Lau, John Locke, Jim Stoddart, Lorenzo Villaggi, Ray Wang, Dale Zhao and David Benjamin, 2016, The Living, Autodesk Studio

Raster-to-Vector: Revisiting Floor Plan Transformation, Chen Liu, Jiajun Wu, Pushmeet Kohli, Yasutaka Furukawa, 2017, Washington University, Deep Mind, MIT

Relational Models for Visual Understanding of Graphical Documents. Application to Architectural Drawings, Llus-Pere de las Heras, 2014, Universitat Autonoma de Barcelona

Shape matching and modeling using skeletal context, Jun Xie, Pheng-Ann Heng, Mubarak Shah, 2007, University of Central Florida, Chinese University of Hong Kong

Statistical segmentation and structural recognition for floor plan interpretation, Lluís-Pere de las Heras, Sheraz Ahmed, Marcus Liwicki, Ernest Valveny, Gemma Sánchez, 2013, Computer Vision Center, Barcelona, Spain

Unsupervised and Notation-Independent Wall Segmentation in Floor Plans Using a Combination of Statistical and Structural Strategies, Lluıs-Pere de las Heras, Ernest Valveny, and Gemma Sanchez, 2014, Computer Vision Center, Barcelona, Spain

Path Planning in Support of Smart Mobility Applications using Generative Adversarial Networks, Mohammadi, Mehdi, Ala Al-Fuqaha, and Jun-Seok Oh. , 2018

Automatic Real-Time Generation of Floor Plans Based on Squarified Treemaps Algorithm, Fernando Marson and Soraia Raupp Musse, 2010, PUCRS

Procedural Modeling of Buildings, Pascal Muller, Peter Wonka, Simon Haegler, Andreas Ulmer, Luc Van Gool, 2015, ETH Zurich, Arizona State University

Generative Design for Architectural Space Planning, Lorenzo Villaggi, and Danil Nagy, 2017, Autodesk Research