Architecture and Affordance:
A Data Driven and Computational Approach to the Architectural Analysis and Design of Buildings Using Affordance as a Model of Typology

By

James Walker

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Architecture (M.Arch)

The Faculty of Graduate Studies
Laurentian University
Sudbury, Ontario, Canada

THESIS DEFENCE COMMITTEE/COMITÉ DE SOUTENANCE DE THÈSE
Laurentian Université/Université Laurentienne
Faculty of Graduate Studies/Faculté des études supérieures

Title of Thesis Titre de la thèse	Architecture and Affordance: A Data Driven and Computational Approach to the Architectural Analysis and Design of Buildings Using Affordance as a Model of Typology
Name of Candidate Nom du candidat	Walker, James
Degree Diplôme	**Master of Architecture (M.Arch)**
Department/Program: Département/Programme	Architecture

Date of Defence: June 22, 2021
Date de la soutenance «Date_of_defence»

APPROVED/APPROUVÉ

Thesis Examiners/Examinateurs de thèse:

Prof. Steven Beites
(Supervisor/Directeur(trice) de thèse)

Prof. Patrick Harrop
(Co-supervisor/Co-directeur(trice) de thèse)

Dr. Ted Krueger
(External Examiner/Examinateur externe)

Approved for the Faculty of Graduate Studies
Approuvé pour la Faculté des études supérieures
Tammy Eger, PhD
Vice-President Research
Vice-rectrice à la recherche
Laurentian University / Université Laurentienne

ACCESSIBILITY CLAUSE AND PERMISSION TO USE

> CONTENTS

Abstract

If you were told to imagine a bedroom, you would make uncountable tiny assumptions about the space in the process. If you asked a computer to imagine a bedroom, it would have no idea where to begin for exactly that reason.

When analyzing space, we can easily fall into the same trap. If a room on a plan is labeled as a bedroom, it's easy to forget that label tells us little about what that space is actually like. How did the designer intend for the user to store things? What kinds of fixtures and furnishings are provided to allow the space to be a bedroom? The main question of this thesis is : How can we systematize the more abstract concepts behind architecture in order to give more power to generative AI techniques?

James Gibson theorizes some useful concepts of how environments afford choice to their users in his book *The Ecological Approach to Visual Perception*. He calls these environmental opportunities "affordances". They are properties defined by the relationship of a person to an object. He argues that instinctively, we understand the world first through utility and then through the invariant properties of objects which give those affordances. This gives us two new tools for the analysis of architecture. First, there is the affordance of space, which acts like architectural program. It defines the goals which the space aims to achieve. Second is the invariants of space, the shapes, forms, and designs of the actual objects which fulfill that function.

Space syntax has long been used as a spatial analysis tool to graphically represent the relationships and connections between spaces. This has been used to promising effect in the past to compare buildings of a similar typology in order to better understand them. We can use the framework of space syntax and introduce affordances to gain a better understanding of how affordances of access, natural light, sound, and activity congregate in building typology. By building a digital sensor array and conducting an analysis of a particular building typology, we can start to find optimal patterns of affordance which exist within living buildings.

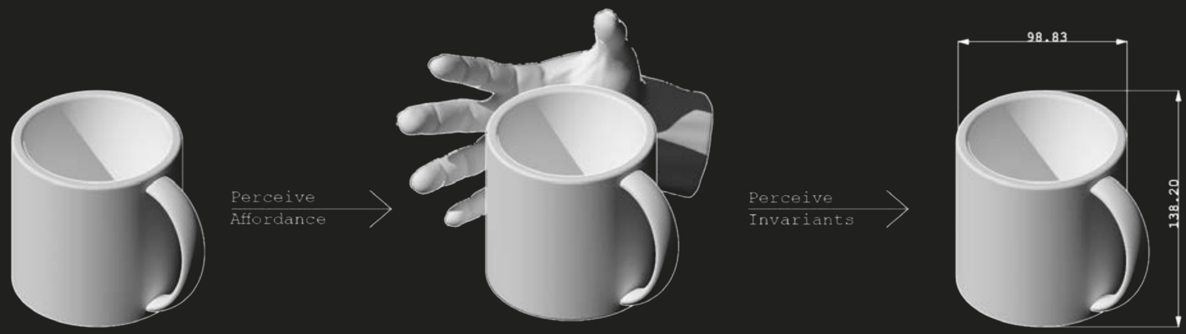


Figure 0.1: Affordances VS Invariant Properties.

The typology I am looking at for this thesis is the United Churches of Sudbury. Sacred space is a phenomenologically dense and interesting typology which lends itself to generating interesting data. United Churches as a typology also have a philosophy of shared multi-use space which lends itself well to this more generalized approach to understanding program through affordances.

In this thesis, I look at four different case studies of United Churches, one to understand what makes a United Church, and three others as examples of typology and subjects for collecting data. The aim of this work is to use this data to develop a set of computational design tools that will eventually be used to suggest a possible design for a United Church on the site of Larchwood Memorial United Church in Dowling.

Aside from site analysis, there are two parts to this affordance based process. Just like how Gibson distinguishes between affordances and the invariant properties of objects, I take stock of affordances through a set of affordance graphs and tables of relevant data from each of my case studies in Sudbury. I also look at the design solutions such as furnishings and building openings which these churches used to satisfy those affordances and document them in the from of a pattern language.

Using the affordance data, I find common patterns of design and layout for a given typology. These patterns of affordance can then be used with a generative algorithm to generate a schematic design. In this thesis, I will present a number of schematic designs and explore the range of outcomes, limitations, and areas for improvement that you can expect from this approach to generative design.

A final schematic design can then be matched with examples of vernacular objects and strategies found and documented through the pattern language of sacred space. Using data that I have collected about openings and furnishings in the case studies, it becomes possible to automatically populate these schematic designs with objects to complete the design. This thesis will detail this process and the theory behind it.

ACKNOWLEDGMENTS

I have received a lot of support and assistance during the writing of this thesis:

I would first like to thank my thesis advisor Professor Steven Beites for his expertise while developing my research questions, creating my methodology, and prioritizing my efforts. His interest and guidance allowed me to probe parts of architectural research that some consider unusual.

I also wish to thank my thesis defense committee, including Professor Patrick Harrop and Dr. Ted Krueger for their insights and guidance regarding my work throughout the development.

I would also like to thank my family and friends for their support and council during the trying times that this work was completed.

Finally, I would like to thank Reverend Susan Peverley and her congregation for graciously allowing me to use their church spaces as case studies for my research. Hopefully the final designs included in this book stoke your imaginations.

FIGURES AND TABLES

0. ABSTRACT

1. A PROBLEM OF DIGITAL REPRESENTATION

2. PREVIOUS GENERATIVE DESIGN TECHNIQUES

A Problem Of Digital Representation

The principal goal of computational design is to break down the creation of an object into sequential steps a computer can complete. Computers, fundamentally can only take numbers from memory, perform mathematical operations on them, and then place them back into memory. This means that when trying to realize the goal of computational design, there is a major caveat; if you want to perform computations on anything that isn't a number, you must first turn that object into a set of numbers.

One abstraction that allows us to make this leap is the "Object". An object can be thought of as a set of values given context. A digital image has a set of numbers representing its size, and then numbers for the red, green, and blue values that make up every individual pixel of the image. Understanding that context, we can recreate or alter the image that the object represents. Everything that we interact with on a computer works in this way. Text is a series of numbers given context to represent characters. Points in space are numbers given context to represent coordinates. Shapes are coordinates given context to represent lines, planes, and boundaries.

While objects work as representations of real life things, just as important are tools that transform objects. It is a requirement to add and delete characters from text boxes, crop and scale images, to translate, rotate, and copy geometry. In a computer, objects exist as sets of data with context, and are transformed into new objects when we apply functions to them. Usually the design of these objects and the functions which can alter them are left up to computer programmers. However, there is a new paradigm of computation which has opened new avenues for development.

Parametric design is the answer to the question: What if the user could define their own objects and transforms? We can make a 3D-model of a door frame given the height and width of the door and the thickness of the frame. With those parameters, the proce-

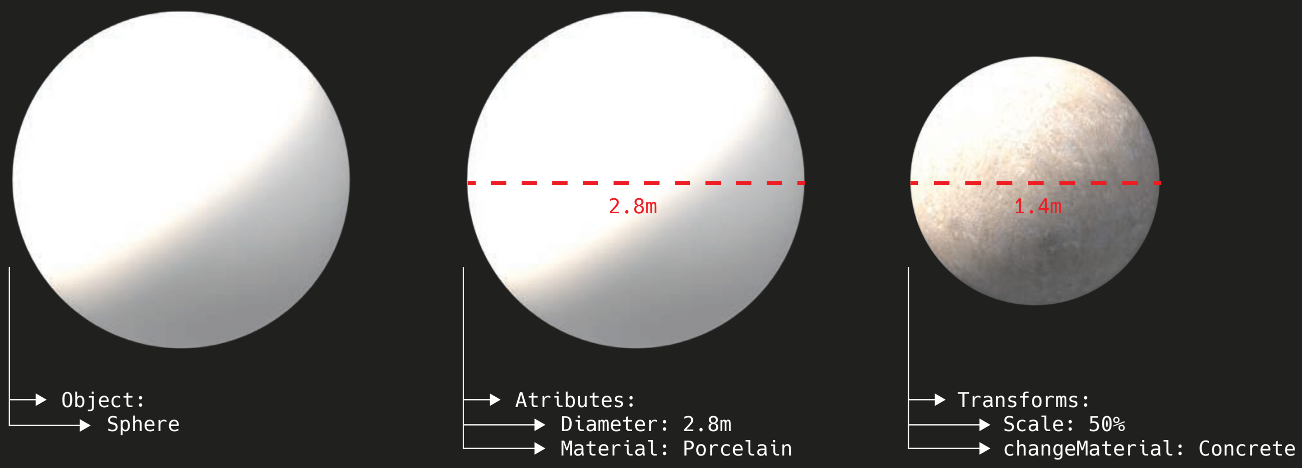


Figure 1.1: Anatomy of a Digital Object (By Author).

dure for modeling the door frame is the same every time, so what if we could make the door frame into an object itself? Walls have a height, length, and thickness. Floors have a perimeter shape and a thickness, What if instead of representing and modeling a building using primitive solids, we abstract the entire thing away into objects that represent individual building components? Parametric design allows us to take the tools, objects, and transformations which we already have, and abstract them away into parameters which in theory are simpler to understand and easier to use.

Understanding a well-designed parametric object and its set of transformations is a fairly painless task. The tough part of parametric design has always been in creating new objects. If there is a class of thing that exists, in order to abstract it into a set of variable parameters, you must first understand the invariant properties of that class that distinguish it from other classes. Inevitably, you must be able to measure these properties and quantify them. This makes certain kinds of abstractions easier to implement than others.

It's simple to understand a suspension bridge as the dimensions and orientations of its cables. We can calculate the effects of forces on this model and optimize the strength of the design or the cost effectiveness of the materials used. This is the core of generative design: taking a parametric object and applying an algorithmic process to it in order to optimize for a particular parameter. Often this is to make the "best" version of an object for a particular task. However, how do you quantify something like the experience of walking across the bridge?

People care about the cultural and psychological effects of objects, and not just their calculable physical properties. We can subjectively compare them, and with some consensus decide when they are good, but it's difficult to say exactly why, or find well-defined parameters which correspond to the features we think are important. This poses a few problems for generative and parametric design going into the future.

First, generative and parametric design are wonderful tools for design problems where variables are entirely controlled for. In mechanical engineering, an elementary task for generative design is finding the form of an object which is the strongest, using the least amount of material. However, in a field like architecture, we aren't always working with variables which have easily calculable answers. This means that when we apply generative design to architecture, we often use it to optimize for things that may not actually generalize into being a good design.

Often when using a generative design process in architecture, we choose to generate forms which optimize for easily understandable variables. We can minimize travel distance between connected programs, we can calculate how much sunlight will make it into a facade, we can optimize for external views. However, we could have optimized for those things fairly well on our own without a computer's help; and we could have done it while considering social, cultural, contextual, and systemic issues which are very difficult to model for a computer. This raises some important questions: Is optimizing for the things we can easily measure actually a good idea? Does using generative design this way actually give us designs which are better than those which grew from a more subjective understanding of the design problem? A computer can take an adjacency diagram and generate 10,000 options for layouts of an office plan, but if buildings are primarily about the experience and interaction between human beings; is that actually a solid basis for an architectural design?

The problem with generative design as it exists today seems to lead back to that problem of digital representation. There are things that human minds can just intuitively understand about architecture which are difficult to represent in a way that a computer can work with. Therefore, the most important question to delve deeper into on this subject seems to be: How can you represent information about culture, the subjective qualities of space, and the human experience in a way that a computer can parse.

Affordance Theory

To start, I looked at phenomenology to better understand the mechanisms by which humans perceive their environment. The theory was relevant, but difficult to apply because much of it was subjective and constructivist in nature. However, the alternative of direct perception is unsuitable because it completely sidesteps the concept of phenomenology altogether, cutting the human experience out of the question entirely. As a strange sort of middle ground *The Ecological Approach to Visual Perception* (Gibson, 1986) proposes a new sort of psychological theory of perception which considers the animal and its environment as a single inseparable pair.[1] The most interesting idea to come out of this book has to be Gibson's Theory of Affordances. He first describes this theory in chapter eight of his book like this:

> The affordances of the environment are what it offers the animal, what it provides or furnishes, either for good or ill. The verb to afford is found in the dictionary, but the noun affordance is not. I have made it up. I mean by it something that refers to both the environment and the animal in a way that no existing term does.[2]

The Theory of Affordances does not focus on the phenomenal aspects of perception. It doesn't seek to explain how we perceive the environment through the sense data of the objects in the environment.[3] Instead, it theorizes that animals usually perceive the environment as a set of what Gibson calls affordances.

An affordance is neither a subjective property of an object nor an objective property, it is a property defined by the relationship of a person to an object. Gibson argues that when looking at an object and deciding what it is and what to do with it, we do not perceive the invariant surfaces and qualities of it and then classify it based on that perception. Instead, we instinctively perceive what actions the object allows us to take within our environment, and it is only later that we can measure which attributes of the object allowed us to take that action.[4]

An example that Gibson uses in his own work to describe this is a seat.[5] We may have many methods by which we can classify an object as a seat, or even the subcategories of a chair, sofa, or bench. However, whether something affords to be a seat has nothing to do with this classification. If an object has a surface that is flat, horizontal, extended, rigid, and exists at knee height above the floor, you are in fact able to sit on it, and therefore it affords the act of sitting. Note that the affordance is both a physical quality of the seat itself, because it is the physical attributes of the seat that allow us to sit; But it is also a subjective quality, the relationship of the perceiver to the object is also important to affordance. We may perceive an object that an adult can sit on easily as too high for a child to climb on top of.[6]

Gibson spends a lot of time describing this theory in terms of objects similar in scale to the human body, but of course we can extend the theory of affordances to the scale of architecture. Architecture is very much the art of providing affordance to the human body. Gibson describes how a vertical, flat, extended surface is a wall or cliff that impedes our locomotion. We can only pass through it if there is a door or gap in the surface some place. He describes how a horizontal, flat, extended, rigid surface acts as a surface of support.[7] We can walk easily across it. Objects like

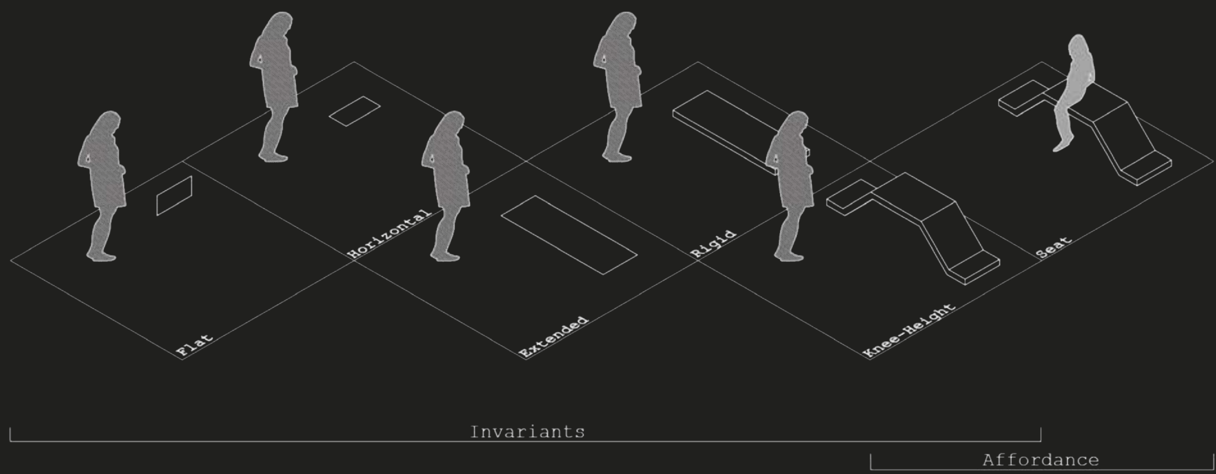


Figure 1.2: The Invariants of a Seat (By Author).

walls can also afford safety, occlusion, and privacy. They block the sun, rain, and wind from reaching us or protect the things important to us.

This theory and many of its examples are framed around the idea of an animal interacting with the natural environment. The idea behind this is that from an ecological perspective, the biological systems which are responsible for perceiving the world are attuned for the natural world by virtue of evolving from and for it. For this thesis, the theory needs to be re-framed in terms of the built environment. For this, it's possible to classify the affordances of architecture into four useful categories.

In Vaughn Michell's paper *The Capability-Affordance Model* (Michell, 2012) he describes a classification of affordance he calls an "Objective Affordance". (Figure 1.3) An objective affordance is an affordance which exists independently of the perception of an agent because of its qualities following the laws of natural science allowing it to exist in a state of equilibrium.[8] A floor's ability to support is an objective affordance due to its state of equilibrium, and a bed's affordance of support exists for the same reason. However, a bed's affordance of sleep is subjective upon the perception of the agent. This distinction goes against the original theory, but he argues for it because the affordance of support of a stable object isn't an affordance which has a possibility of happening, it's an affordance which is already being actively performed in the environment. We use objects like walls, floors, and roofs in this way, as stable solid objects that can support our weight and block unwanted stimuli from entering our vicinity. These stable objects with "Objective Affordances" are the first type of affordance used in architecture and they are the grammar which we use to define the boundaries between the built environment and the natural one.

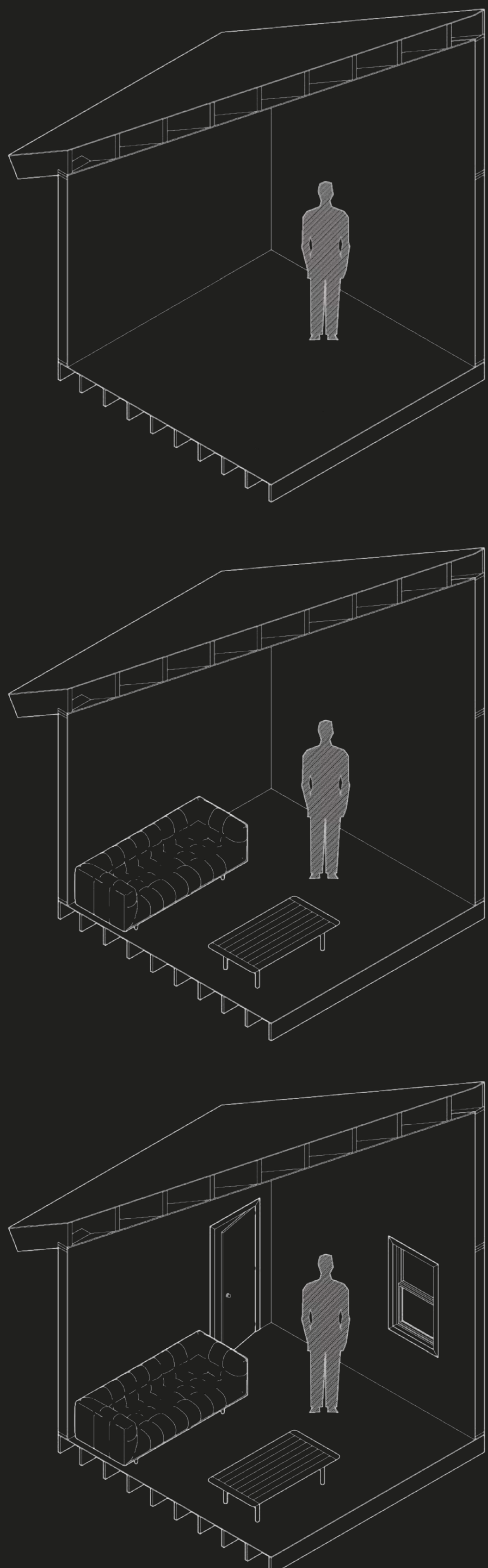


Figure 1.3: Objective and Active Affordances (By Author).

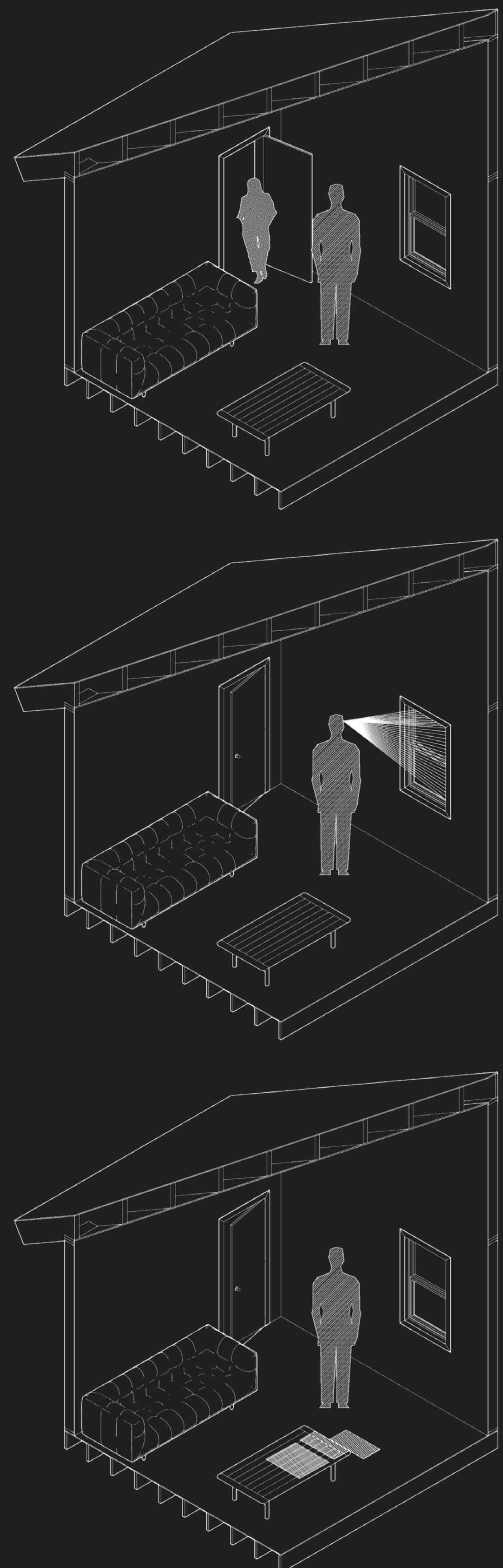


Figure 1.4: Environmental Affordances (By Author).

The second architectural affordance is what I’m calling the “Active Affordance”. (Figure 1.3) This affordance is the traditional type which exists as a relationship between an agent and an object. In architecture, we would describe it as the relationship between a person and the furniture in a space. Active affordances give a sense of purpose to a space. For example, a bedroom is a bedroom because it affords the ability to sleep and the ability to keep one’s belongings safe. An office is an office because it affords you the ability to work.

A special type of active affordance is what I’m calling an “Interface”. Interfaces are active affordances that sit on top of and inside of objective affordances and give agents the ability to alter the environment. Objective affordances do a superb job of impeding various active forces in the environment and changing the experience of a space from a natural one to a constructed one, but they cannot do this job perfectly, and quite often we design them not to. A window is an interface in a wall that affords the ability to allow natural light and air into a room. If there is clear glass in the window, or the blinds aren’t closed, it affords the ability to see out into the landscape. A door in a wall affords access into the adjoining room, or the ability to lock a room from access to others. These interfaces allow interaction with the environment through active affordances.

The final, type of affordance is what I’m calling an environmental affordance.
(Figure 1.4) Environmental affordances are the affordances granted by what the interface allows to enter the space. Natural light affords the ability to see. Fresh air affords a change of temperature in a room, and more oxygen to the brain. A view out a window, or sound passing through a wall affords the ability to act upon information in an adjoining environment. Walking through a door affords interaction with other sets of affordances in other spaces.

Adding these three types of affordances together, it’s possible to create a definition of architectural space framed through the theory of affordances. A singular architectural space is a place that through a combination of active, objective, and environmental affordances, independent of manipulation of the environment, and time, offers a consistent set of affordances to a given agent.

The quality of affordances which places them in between phenomenological objects and physical objects makes them an interesting device to use when studying space. They are the perception of a human being quantized into a single data point and reflected in the space. They are measured aspects of space given the perspective of human perception and it is that aspect that makes them useful for transforming the subjective qualities of architectural space for the digestion of computers.

Space Syntax

A very common body of theory to use as a baseline in this research area is Space Syntax. There is almost no paper on computational architecture analysis which doesn't cite the work done by Bill Hillier and Julienne Hanson in *The Social Logic of Space* (Hillier, 1984).

In this book, they criticize the previous body of work done towards defining spatial systems, saying that they don't succeed in creating a systematic approach by which we can analyze space.[9] They were interested in creating a mathematically rigorous theory of architecture and anthropology that could predict and explain how social interactions between people lead directly to certain types of built forms and spatial grammars.[10]

The core of this research is in solving what the authors call "The Problem of Space". A solution to the connection between human society and the space which makes it up.[11] The solution is presented as the encoding of space into a type of discrete system which describes the morphological features of the architectural space while simultaneously encoding information about the relational identity of the space within the larger system.[12]

The definition of these discrete systems are defined at two different scales: At the scale of the city and at the scale of the individual building.[13] From these two different scales, they developed two primary methods of spatial analysis called alpha-analysis and gamma-analysis.

Alpha analysis is catered to the more open spaces of a city street. The main difference between this type of analysis and the Gamma-Analysis of an enclosed space, is the requirement to turn the continuous open space of a city street into a discrete graphical system.[14] This works by turning the open spaces into a set of convex spaces and axial lines which pass through them. (Figure 1.5)

Gamma analysis works in much the same way, but instead of having to grapple with the open space of streets, it is more simply able to divide space into rooms and the doors which connect them and allow movement between them.[15]

These systems create connected graphs which allow for the creation of various classifications of certain spatial configurations, and analytical methods by which you could determine the properties of subspaces within a given spatial system. The nodes of the graph represent spatial locations which have a certain function or purpose in the space, and the edges of the graph represent connections of access between these spaces.

Graphs like this have an inherent topology to them, and a hierarchy. The use of these graphs is that this topology of space, and values that can be calculated from it can give us insight into the social function of these spaces.

There are three main concepts of measurement which are thought to be important for understanding these systems: Integration, control, and depth. Integration is the average depth of space to all other spaces in a system. A particularly integrated point is exactly one step away from all other points in a system, and a point deep in the system is sufficiently far away from many other points.[16] Integration therefore denotes the privacy or publicity of a space.[17] Control measures how much a space controls access to neighboring spaces and participates in the connectivity of one particular path through the space to another.[18] Depth is a measure which is more important to understanding the interior of a closed system. It is the distance from a given point to the entrance or exit of the system. It functions similarly to integration in that it is capable of measuring the distance and privacy of space in a building, but is more important when considering the boundary of a system. For example, a building has controlled points of entrance and exit, and therefore the depth of a given space from that boundary can be a different measure of the integration of a space.[19]

This system of measurement details a powerful set of tools for the representation and analysis of the topology of space. The general concept of graphical analysis like this can also be extended to other types of data, and is applicable to the measurement and movement of environmental affordances.

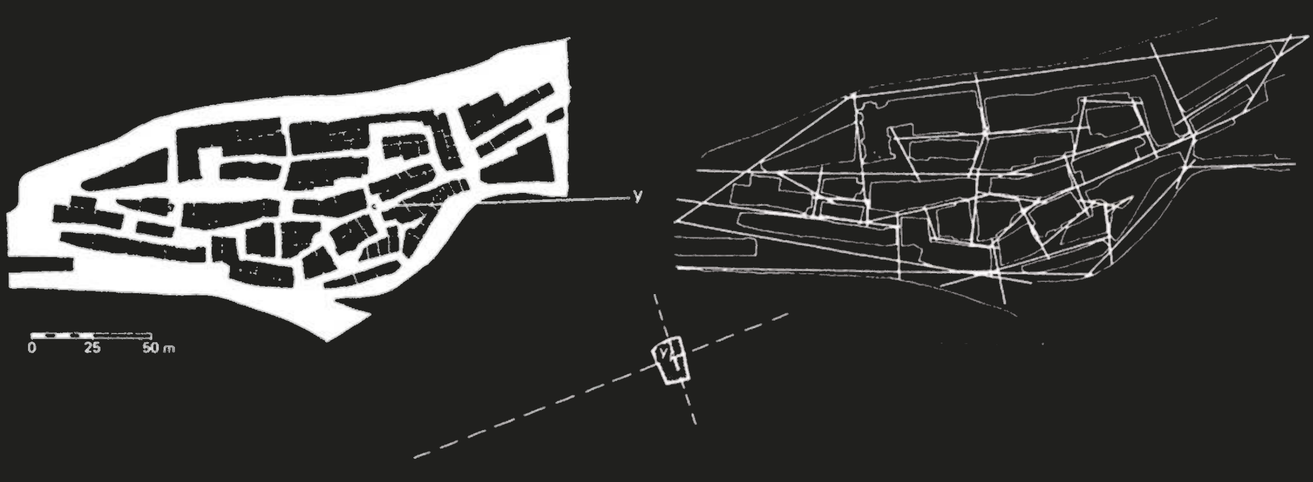


Figure 1.5: Convex and Axial Mapping of the Small Town of G used in Alpha Analysis (Hillier and Hanson).

Pattern Languages

Space syntax is not well suited to dealing with the subjectivity and sensuality of space. Two buildings may result in the exact same graphical analysis, but still have elements within and between the vertexes of the graph that completely alter the use of the space. There are cultural contexts that can alter the meaning of space syntax and what space actually means for those interacting with it. Space syntax is simply a tool which requires context and informed methodology to get useful results from.

One of the ways I hope to supplement space syntax analysis is with affordance analysis as a more generalized approach to the idea of space usage and the flow of information through a building. However, there is also the problem of sensuality, materiality, and detail. If there is a space syntax model of a building representing what the building affords the environment, how could you reverse this process and reintroduce the invariant surfaces and forms back into the system to allow for generation?

The book *A Pattern Language* (Alexander, 1977) is a dictionary of what Alexander calls “Patterns”, or design solutions for a particular problem. Each item of vocabulary also comes with a particular syntax, or context which will direct its use and also suggest other design solutions.[20] The idea is that given a design problem, it would be possible to select a series of solutions from the vocabulary which satisfy it and also have a set of adjacent solutions presented through the syntax. (Figure 1.6)[21]

Beyond the specific goals of the originating text, generally speaking, a pattern language is a hierarchical set of systems organized by size in order of which systems contain other systems. This is why it has been adopted by so many other fields. When analyzing a system of discrete objects, we can start by defining the super-classes of object and work our way down from there, analyzing the various sub-classes which comprise the larger objects until we arrive at the very most elemental components of our design system. Applying this to architecture and affordances, we can deconstruct architecture from a building, to the spaces which make it up, to the affordances in those spaces, to the objects which provide those affordances, to the invariant qualities which provide those affordances, and so on.

Looking at buildings as a super-class, a sub-class of space might be the expected affordance of a room as a whole; a bedroom, kitchen, hallway, or study. Within each of those classes are expected active affordances that need to be accessible in order to fulfill the requirements of their respective super-class; a place to sleep, a source of heat for cooking, an unobstructed walkway, or storage space for books. Each of these affordances in turn could also have differing invariants which allow for their perceived uses and have different benefits for accessibility, effectiveness, and sensual experience.

Deconstructing a number of given buildings in a particular typology in this way can provide a deeper understand how it functions and what trends of design exist within the typology. Unlike Alexander's book *A Pattern Language* which attempts to capture an eternal and universal way of organizing the world,[22] a pattern language can capture the sensual, material, and functional invariants of the affordances inherent to a specific building typology. This can then be used to reconstruct the material and tangable parts of what is lost in the translation of actual space to affordances and graphs.

Space Syntax, and affordance theory together define space in a broad sense, but it is also important to document the specificity in the design solutions encountered so that later they can be recalled and used when creating and designing new iterations of the typology.

Pattern: A Drinking Glass

Problem: It's inconvenient to have to consume liquids only at the source that they come from.

Solution: Liquids can be transported over longer distances within hollow objects with walls that help them form a solid shape.

Syntax: The object should be movable
-- *see Portable Objects*,
Able to contain liquids
-- *see Airtight Containers*,
Visibility may be desired
-- *see Transparent Objects*,
Must be resistant to spilling
-- *see Bottom Heavy Objects*

Figure 1.6: Diagram of a Simple Pattern (By Author).

Endnotes

1.) James J. Gibson, The Ecological Approach to Visual Perception (New York and London: Psychology Press, 2015), p.36.

2.) Ibid. p.177.

3.) Ibid. p.190.

4.) Ibid.

5.) Ibid. p.178.

6.) Ibid.

7.) Ibid. p.183.

8.) Mitchell Vaughan, “The Capability-affordance Model: A Method for analysis and Modelling of Capabilities and Affordances,” in Proceedings of the Second International Symposium on Business Modeling and Software Design (Science and Technology Publications, 2012), p.61-62, accessed November 13, 2020, https://www.scitepress.org/papers/2012/44612/44612.pdf)

9.) Bill Hillier and Julienne Hanson, *The Social Logic of Space* (Cambridge University Press, 1984), p.6)

10.) Ibid. p.27.

11.) Ibid. p.26.

12.) Ibid. p.48-51.

13.) Ibid. p.90.

14.) Ibid.

15.) Ibid. p.147-149.

16.) Ibid. p.108.

17.) Ibid. p.96.

18.) Ibid.

19.) Ibid. p.149-152.

20.) Christopher Alexander, Sara Tshikawa, and Murray Silverstein, *A Pattern Language* (New York: Oxford University Press, 1977), p.x-xi.

21.) Ibid. p.xii.

22.) Ibid. p.7.

Previous Generative Design Techniques

Other researchers have explored the problem of generative design and their work is relevant to understanding the approach and implementation of generative design in this thesis.

Humanizing the Computational Design Process and Space Syntax Variables (Al-Jokhadar, 2016) uses Space Syntax and Shape Grammars to look at courtyard houses from the Middle East and North Africa as a typology.[1] They identify the types of space that can be found in this typology and then apply an altered Space Syntax analysis to their case studies which measure size and shape of the rooms in the building, depth and hierarchy, solar orientation, distance between the centers of the spaces, and whether the space is gendered or not.[2]

In analyzing the variables about the spaces inside large number of buildings within that particular typology, they were able to find patterns that connected all the buildings of that type together and design a Shape Grammar that they thought would be able to generate more designs of buildings within the same type.[3]

This is an example of an approach supplementing Space Syntax to look for patterns in a particular typology of building. (Figure 2.1)

Computer Generated Residential Building Layouts (Merrell, 2010) is a generative design paper using two novel and uncommon approaches to modeling space. The Authors used a Baysian network to represent the probability of certain choices being made in the generation of the spacial syntax diagram.[4] This would allow them to make choices about the kind of building they wanted to generate beforehand, then use chance to fill in the rest of the network.

The network was trained on 120 human designed architectural programs in order to learn the probability of connectivity between certain spaces, the area of each type of space, and the type of connection between spaces, given previous information about the design.[5] (Figure 2.2)

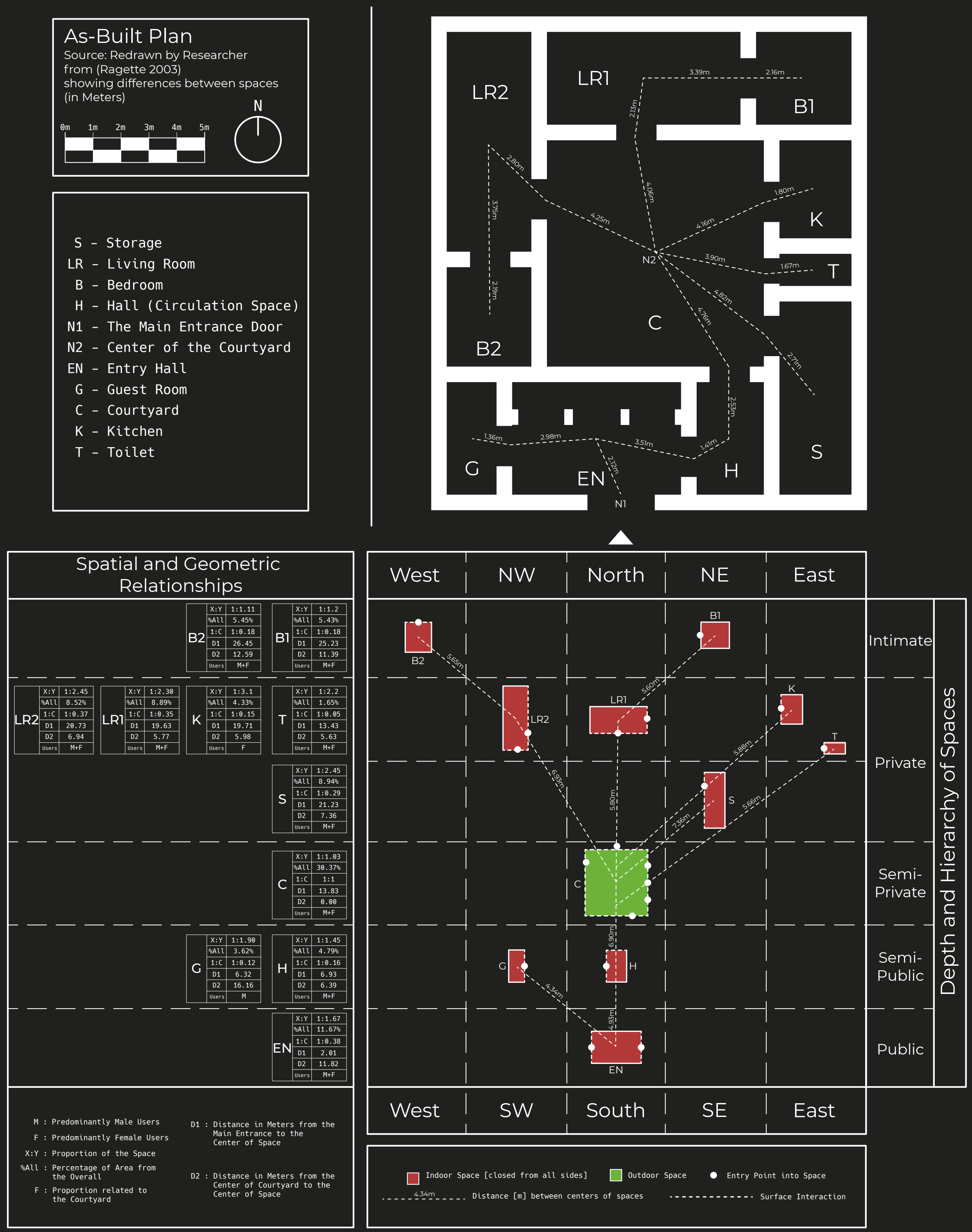


Figure 2.1: A Hybrid Approach to Graphical Analysis.
(Al-Jokhadar, 2016, re-drawn by author)

After the generation of adjacencies and room areas, the next step is optimization. The building is initialized with all the rooms in a grid with the exact same size and then the building is optimized from there.[6] Optimization is accomplished by applying a set of simple transformations. A wall is either slid along a perpendicular plane a certain amount, or two room labels are swapped with one another to simulate a different adjacency.[7] The cost function to evaluate these moves takes into account the accessibility and adjacency of spaces, the dimensions of the spaces, whether the spaces fit into an appropriate footprint, and whether the shape of the room is convex.[8]

This is an example of the statistical analysis of the connections between space using graphical analysis in order to generate architecture.

Artificial Intelligence in Architecture: Generating Conceptual Design via Deep Learning (As, 2018) is a paper which harnesses the power of deep neural networks to analyze and generate residential architectural designs. Very similarly to the direction which I am planning to take my research in, the authors of this paper took a graph based approach to modeling architecture for a neural network. Nodes in their graphs represent "type, area, volume, and perimeter" and edges represent "the type of adjacencies between rooms".[9] They also note that graphs have the ability to represent other types of information about rooms, like furniture and objects within the space.[10]

Next, the training dataset of fifteen building was subjectively evaluated on a set of criteria including "Livability" and "Sleepability". The purpose of the deep neural network is to determine which parts of the building correspond most strongly with these subjective values and output them. Doing this creates a set of smaller functional elements of a building which can be recombined into new designs.[11] (Figure 2.3)

These nodes were then recombined using graph merging algorithms and checked to see if they were missing any program elements which would normally be seen in a residential design. These were then added back into the graph by a DNN network that had been trained to add a single room to a pre-existing network while making sure that room adjacencies were consistent with the original dataset.[12]

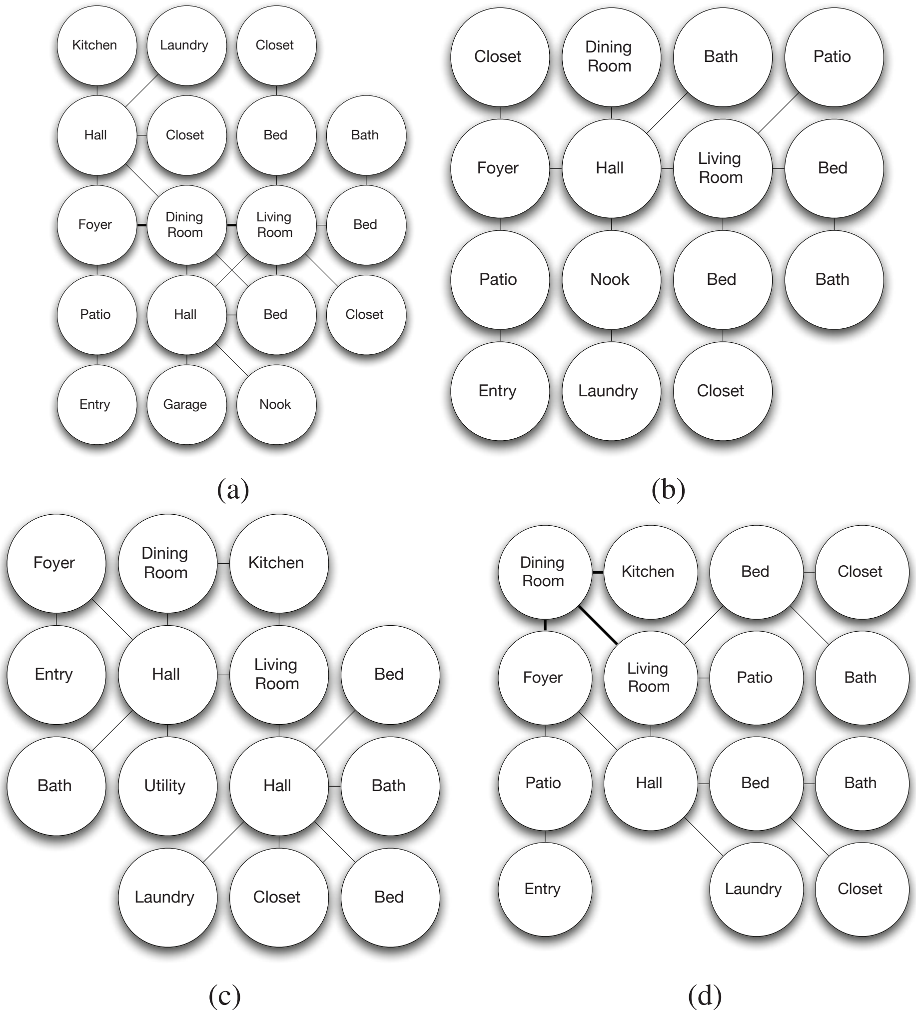


Figure 2.2: Generated Graphs from Computer Generated Residential Building Layouts (Merrell, 2010).

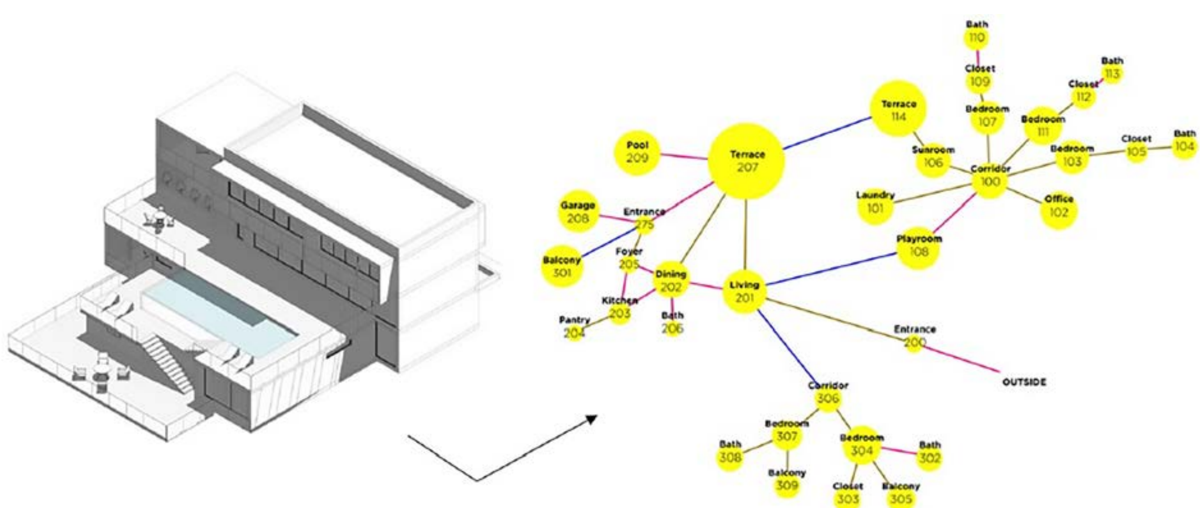


Figure 2.3: Artificial Intelligence in Architecture (As, 2018).

Generative Approach

The approach taken in this thesis for a generative system follows pretty closely from *Computer Generated Residential Building layouts* (Merrell, 2010). This paper uses a simple shape grammar to randomly manipulate the boundaries of a floor plan in an attempt to minimize a set of cost functions. Regarding the possibility of learning relations in data using a baysian network, the data set that can be collected in a reasonable amount of time will be too small for that approach to be feasible. Instead, the approach is to capture qualitative data similar to the types of data captured in *Humanizing the Computational Design Process and Space Syntax Variables* (Al-Jokhadar, 2016) and then use that data during the optimization process to create upper and lower boundaries for certain variables of the cost function.

The goal of this generative algorithm is to generate a building massing which is compatible with the requirements of the given affordance data. This is achieved using a simulated annealing optimization algorithm with six steps overall.

The first step is to generate the most basic form of massing with square rooms of an equal height. Along with this, all the data that is needed for the calculation of the cost function like the adjacency matrix and upper and lower bounds on the sizes and shapes of rooms should be entered. Next, the initial floor plan is passed to the slide transform which will choose a random room and wall and then slide it a random amount in a random direction and also randomly change the height of one of the spaces in the building. Then, the floor plan is passed to the swap function; two random rooms are chosen and their labels are swapped with each other. The original floor plan, and the transformed floor plan are then evaluated using a cost function that takes into account how well the floor plans fit six different attributes that are directly related to the data collected. The floor area of each space, adjacency between spaces, ratio of length to width, convexity or concavity of the spaces, height of the spaces, and the amount of surface area exposed to sun are all taken into account for the generation of massing. Finally, a function looks at the results of these two cost functions and probabilistically decides if the transformations should be accepted or rejected. If the function determines that the changes result in a decrease of the cost function or an increase which doesn’t fall too far outside the annealing value, the floor plan is accepted and the cycle continues with the new transforms. Otherwise, the old floor plan is restored, and the cycle continues from that point instead. (Figure 2.4)

In order for the algorithm to function, we need to provide data about the type of massing

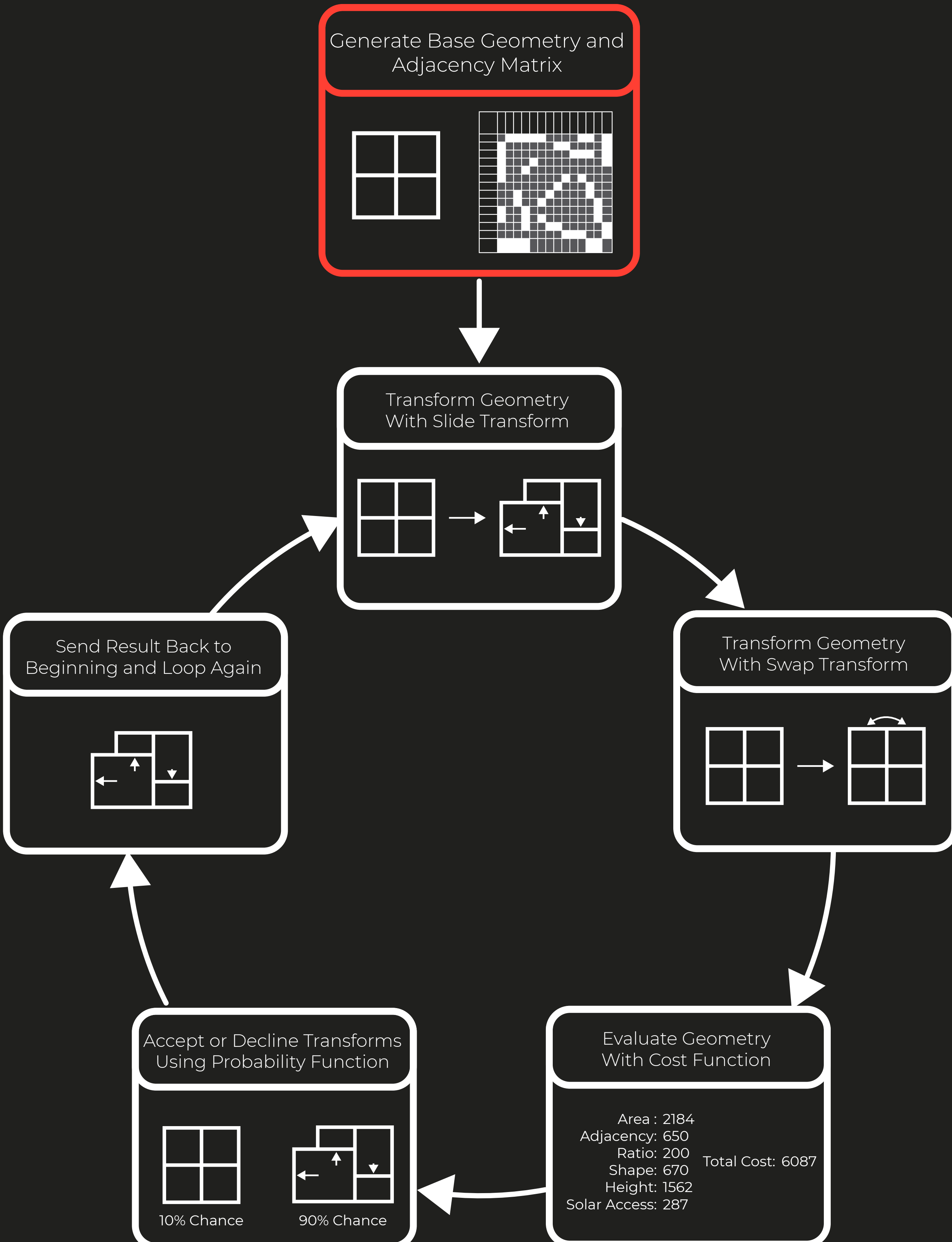


Figure 2.4: Overview of Generative Algorithm (By Author).

that we want to generate. For most cost variables, an upper and lower bound is given for each space within the building as a whole. The purpose of the cost function is to determine the fitness of a given floor plan option in order to compare it to other options. Typically, the lower the number is the better it is, and the goal is to bring this cost value down to zero over time.

For the area cost, if the space fits within the upper and lower bounds specified it doesn't increase the cost function, however if a space is outside of the bounds it increases the cost function by however far outside it is. (Figure 2.5) Through iterative changes over time, the algorithm tries to minimize the cost function and bring them all down to zero.

For the adjacency cost, the algorithm is supplied with data about which rooms should and shouldn't be touching. This information takes the form of an adjacency matrix. Given this matrix, the function then calculates the theoretical maximum score that a floor plan could achieve using this matrix. To do this, it adds all the blue squares in the matrix together and then divides by two. The base cost for the example matrix is eleven. (Figure 2.5) If two spaces that should be adjacent to each other are adjacent, the algorithm subtracts one from the base cost, if two spaces that shouldn't be touching are adjacent, it adds one to the base cost. Through iterative changes, a local minimum of adjacency can be found. (Figure 2.5)

For the ratio cost function, the algorithm is given the upper and lower bound for the ratio of a space's length to its width. It draws a bounding box around the space in order to get a length and width from irregular shapes and then compares this ratio to the upper and lower bounds supplied from the data. This introduces a problem however; the algorithm may be incentivized to create long skinny protrusions from itself in order to minimize this cost. To solve this, as an additional penalty to irregularly shaped rooms , the program also adds the distance between the bounding box's center of mass and the room's center of mass to the cost, that way creating irregularly shaped rooms without a good reason will boost the cost function. (Figure 2.5)

In buildings typically it is preferred that spaces are convex. Convex spaces are spaces where the whole space is visible from any one point within it. Sometimes there are benefits to concave space, like privacy or adjacency, however it is beneficial to provide an incentive to keep spaces convex so that concave spaces are only created when there is a concrete benefit to another cost value. We can calculate the convexity of a given space by connecting all the vertices of a space with a line and then finding equally spaced points along these lines. A perfectly convex space will never have any points on the outside of the curve, and the more convex the space is, the less points will be on the outside of the shape. The shape cost of the rooms in a floor plan is

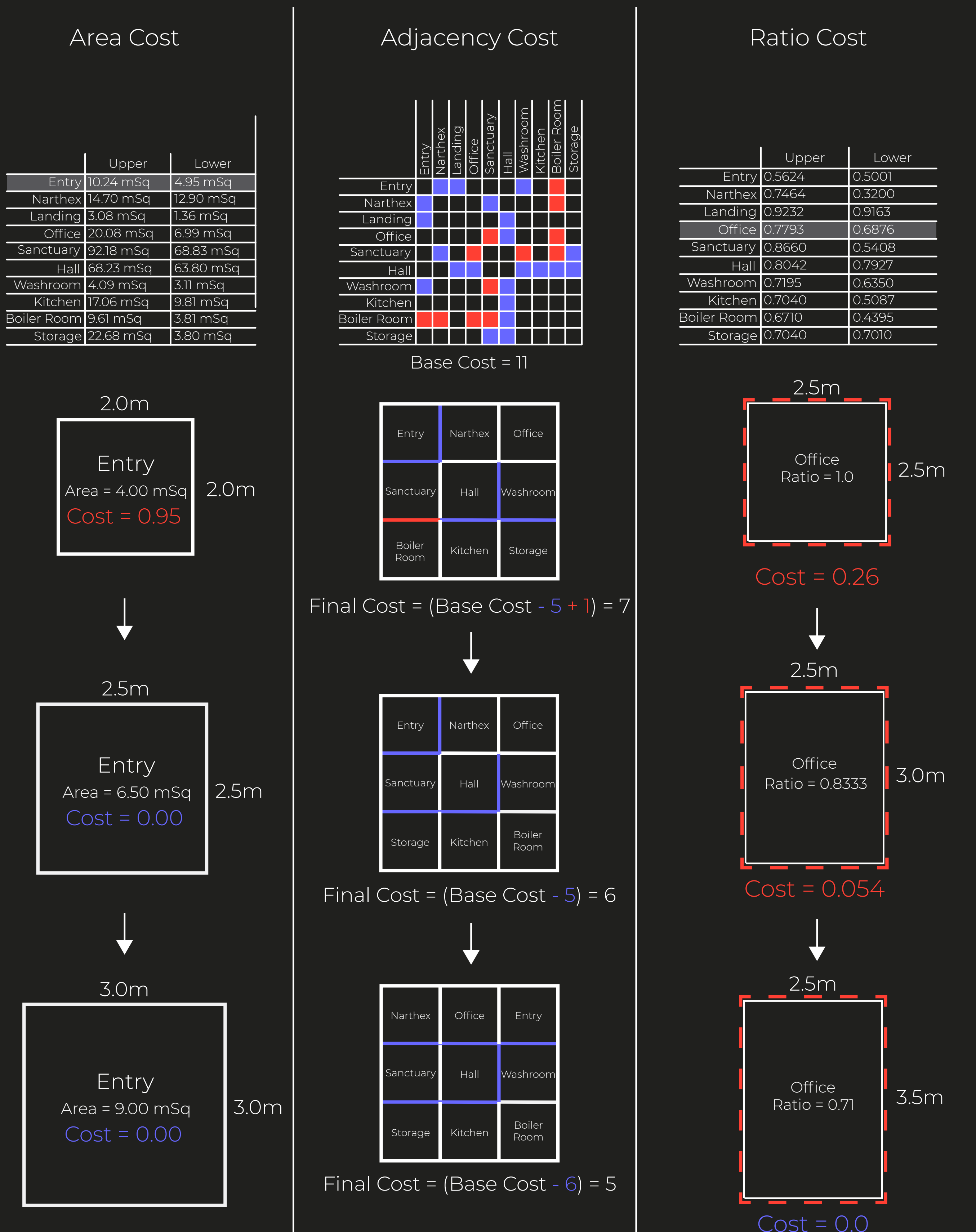


Figure 2.5: Area, Adjacency, and Ratio Cost Function Overview (By Author).

therefore the amount of points which are outside of the space after this calculation. (Figure 2.7)

Similar to area, height can also be optimized using data on the upper and lower bounds of the heights of spaces within a building. The algorithm treats the floor plan like a massing model and extrudes the rooms to a given height value. The height cost is how far outside of these bounds the height values of the floor plan happen to be. (Figure 2.7)

Finally, the solar access cost calculates the amount of solar exposure that the walls of a space get at different times of the day. A lower bound for the required solar exposure can be calculated using the average glazed surface area found from each space. A ray-casting algorithm then estimates how much surface area of the building is exposed to direct sunlight, and if a space isn't getting enough sunlight this function increases the cost value. (Figure 2.7)

The results of all six of the different cost functions are brought together in a final cost function. (Figure 2.6) Each cost value is multiplied by a weight and then added together. This gives us a fine control over how much the algorithm takes each cost into account and can let the user or algorithm re-balance this process on the fly. In this example, biases in how the cost functions are calculated are balanced out so each part of the cost function is given equal weight.

The last part of the generative algorithm is the selection function where the massing model, before and after the transformations, has their costs evaluated. The selection function compares the cost scores of both models and the result is a probability that the transforms will be accepted over the original. If the cost value of the transformed plan is lower than the original cost, the probability of being accepted is 100%. Otherwise, the closer the new cost value is to the old cost value, the more likely the plan is to be accepted. How much the probability is restricted is determined by the annealing value. The higher the value, the more restrictive the selection function is. This simulated annealing will find local maximums in the fitness space over time. (Figure 2.6)

	Area Cost	Adjacency Cost	Ratio Cost	Shape Cost	Height Cost	Solar Cost
Cost	2.01	0.834	11	2.0	0.3	0.78
Weight	0.15	0.36	0.03	0.15	1.0	0.38

$$(W_{area} * C_{area}) + (W_{adj} * C_{adj}) + (W_{ratio} * C_{ratio}) + (W_{shape} * C_{shape}) + (W_{height} * C_{height}) + (W_{solar} * C_{solar}) = Final\_Cost$$

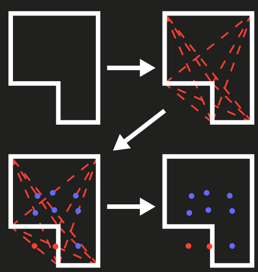
$$(0.15 * 2.01) + (0.36 * 0.834) + (0.03 * 11) + (0.15 * 2) + (1.0 * 0.3) + (0.38 * 0.78) = Final\_Cost$$

$$0.301 + 0.300 + 0.33 + 0.3 + 0.3 + 0.296 + 0.294 = 2.121$$

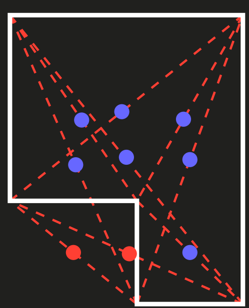
$$\min(1, \exp(Annealing * (Old\_Cost - Final\_Cost))) = Selection\_Probability$$

Figure 2.6: Generative Algorithm Selection Function (By Author).

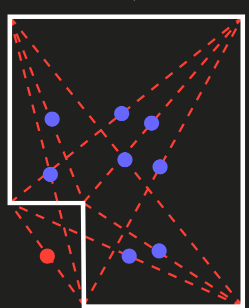
Shape Cost



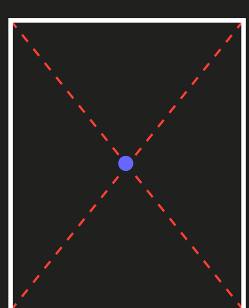
Points Inside = 7
Points Outside = 2



Cost = 2



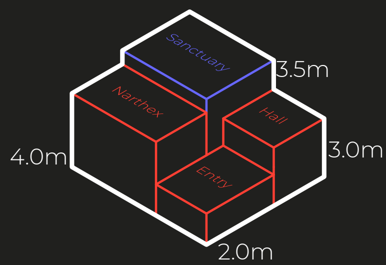
Cost = 1



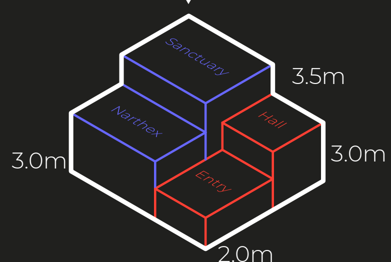
Cost = 0

Height Cost

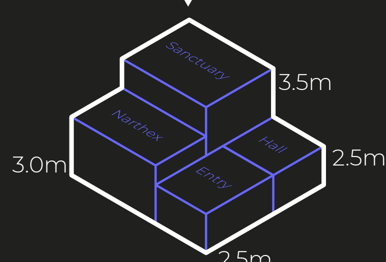
	Upper	Lower
Entry	4.16 m	2.41 m
Narthex	3.27 m	2.10 m
Landing	4.06 m	2.26 m
Office	2.64 m	2.41 m
Sanctuary	4.52 m	2.87 m
Hall	2.69 m	2.26 m
Washroom	2.26 m	2.26 m
Kitchen	2.69 m	2.26 m
Boiler Room	2.43 m	1.52 m
Storage	2.41 m	1.93 m



Sanctuary Cost = 0.0
Hall Cost = 0.3
Entry Cost = 0.4
Narthex Cost = 0.7



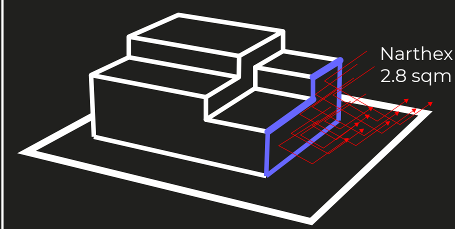
Sanctuary Cost = 0.0
Hall Cost = 0.3
Entry Cost = 0.4
Narthex Cost = 0.0



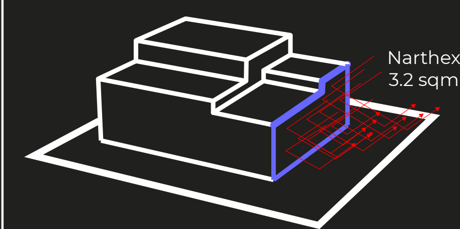
Sanctuary Cost = 0.0
Hall Cost = 0.0
Entry Cost = 0.0
Narthex Cost = 0.0

Solar Access Cost

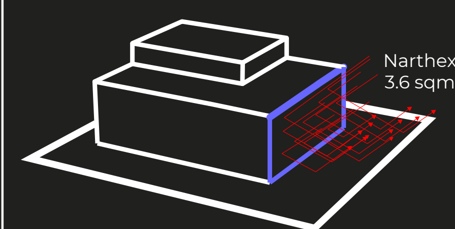
	Lower
Entry	1.61 mSq
Narthex	3.58 m
Landing	0.00 m
Office	1.11 m
Sanctuary	4.71 m
Hall	4.22 m
Washroom	0.00 m
Kitchen	0.00 m
Boiler Room	0.00 m
Storage	0.00 m



Cost = 0.78



Cost = 0.38



Cost = 0.0

Figure 2.7: Shape, Height, and Solar Access Cost Function Overview (By Author).

Data Collection Methods

Data is captured and stored in spreadsheets which represent complex graphs. Space is modeled as a node on this graph which has inherent static qualities associated with it. These qualities include things like active affordances of the room, the general room designation, the area and volume of the room, height of the room, the room proportions, room temperature, average brightness, background noise level, etc. Edges between these nodes represent building interfaces such as doors, windows, stairs, elevators, openings, etc. Each of these edges representing a building interface will also have values associated with them. These edges will encode information such as the number of stairs, change in elevation, the dimensions of an opening, the height from the floor of an opening, if it's possible to pass through an opening, if it's possible to see through an opening, if an opening lets light through it, now much light an opening lets through it, if an opening has a door on it, if a door has a lock on it, etc.

The exact variables that were tested for and their units can be found in Tables 2.1, 2.2, and 2.3.

Because the intensity of light varies depending on the time of day and year, it is necessary to control for this. The method I propose is to first measure the external illuminance from the sun on each side of the building prior to measuring the inside. Then by measuring the illuminance inside the building at a distance of one meter away from the window, we can get a measure of the difference between the quality of light before and after. (Figure 2.8) This value is what is referred to in the tables as the relative illuminance of a given opening.

In many ways this is a naive approach to capturing information about how light passes through these building interfaces. Reducing the way that energy interacts with a building to a few variables can't fully encompass the physics involved, but this approach should give us better context for the kinds of interfaces which mediate the spaces in a building without requiring specialized equipment.

By collecting this data, it is then possible to graphically represent the connections and relationships between spaces in a building. With a large enough data set of information about connections between space, light, sound, and types of interfaces, researchers could then make predictions about what kinds of environmental affordances are common and required for certain active affordances to take place in the space.

As a basic example, if we were to take a large enough data set of the number of windows and the average light level in a living space it would be possible

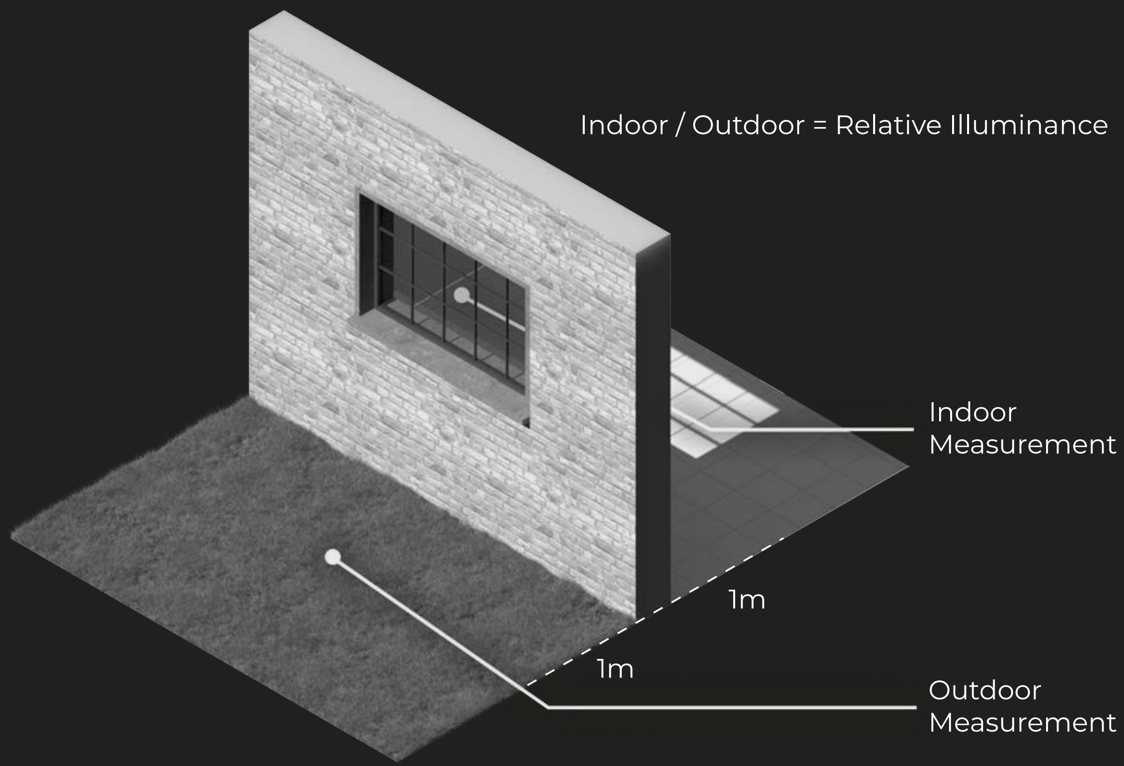


Figure 2.8: Measuring Relative Illuminance (By Author).

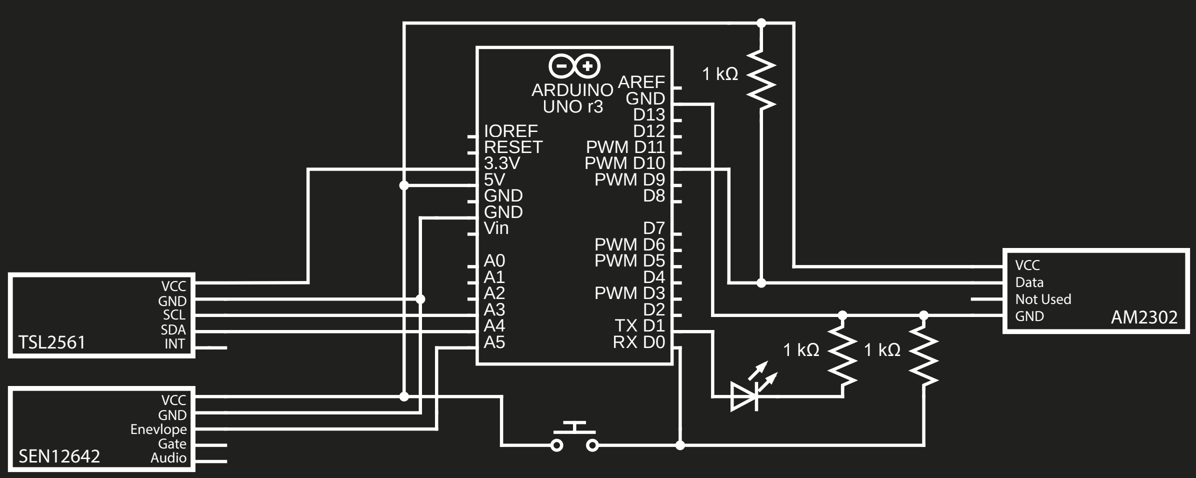


Figure 2.9: Circuit Diagram for Data Capture Card (By Author).

to construct a histogram of that information and create a probability distribution of the range of possible options. I could make the choice as a designer to act on intuition using this information, or encode it as a probability distribution and allow a computer to choose a value itself from that probability space.

Data about the light, sound and temperature levels was captured, through site visits, using a digital sensor array. (Figure 2.9) There are three sensors used in total. The first sensor is the AM2302, which is a digital humidity and temperature sensor. It uses a capacitive humidity sensor and a thermistor to measure the environment it’s placed in, and output the information digitally over a serial pin. The second sensor is the TSL2561. It uses two photo-diodes to measure the total illuminance of the light falling onto it taking infrared radiation into account. This is also output digitally over serial. Finally, the array uses the SEN12642 sound sensor to measure the amplitude of sound coming from the environment. It has a microphone and amplifier which allows it to output the audio, an analog representation of the audio amplitude, and a digital indication of the presence of sound.

These sensors are hooked up to the inputs of an Arduino UNO which is in turn attached to an ADAFRUIT data logging shield. An attached button tells the Arduino to poll the sensor array and save the information to the SD card. An LED located on the board also gives feedback as to whether the button has been pushed and if the arduino has finished collecting valid data from the sensors. This device will allows information to be captured about various architectural environments on location, and store that information on an SD card for later review.

Global Variables		
Features	Variables	
Surrounding Environment	v1	External tempertaure at time of recording <C>
	v2	Northern external illuminance at time of recording <lux>
	v3	Eastern external illuminance at time of recording <lux>
	v4	Southern external illuminance at time of recording <lux>
	v5	Western external illuminance at time of recording <lux>
Building Features	v6	Expected building occupancy <# of people>
	v7	Number of floors <number>
	v8	Average building temperature <C>
	v9	Average building illuminance <lux>
	v10	Average building background noise <dB>
Overall Geometry	v8	Footprint width <m>
	v9	Footprint height <m>
	v10	Footprint area <m^2>

Table 2.1: Network Wide Variables (By Author).

Node Variables		
Features	Variables	
Description	v11	Name of Interior space <identifier>
Active Affordances	v12	Sitting <boolean>
	v13	Sleeping <boolean>
	v14	Cooking <boolean>
	v15	Eating <boolean>
	v16	Relaxing <boolean>
	v17	Meeting <boolean>
	vXX	...Continue with affordances relevant to your typology
Geometric Properties	v18	Space width <m>
	v19	Space length <m>
	v20	Space height <m>
	v21	Space area <m^2>
	v22	Space aspect ratio <1:X>
	v23	Percentage of the space relative to footprint <m^2>
Environmental Properties	v24	Average illuminance relative to outdoors <lux_internal / lux_external>
	v25	Background noise <dB>
	v26	Average temperature <C>
Material Properties	v27	Floor material <Wood, stone, tile, carpet, concrete...>
	v28	Wall material <Wood, stone, tile, carpet, concrete, drywall...>
	v29	Ceiling material <Wood, stone, tile, carpet, concrete, drywall...>
Space Syntax Variables	v30	Integration value of space <number>
	v31	Depth value of space <number>

Table 2.2: Variables for each network node (By Author).

Edge Variables		
Features	Variables	
Description	v32	Type of interface <Door, window, stairs, elevator, stairlift...>
Environmental Affordances	v33	Interface allows light to pass through <boolean>
	v34	Is possible to see through interface <boolean>
	v35	Is possible to travel through interface <boolean>
	v36	Interface can be toggled <boolean>
	v37	Relative illumnance before toggling <lux_internal / lux_external>
	v38	Relative illuminance after toggling <lux_internal / lux_external>
	v39	Temperature adjacent to interface
	v40	Change in elevation through interface <m>
	v41	Interface has door/covering <boolean>
	v42	Interface is lockable <boolean>
Geometric Properties	v43	Interface height <m>
	v44	Interface width <m>
	v45	Interface depth <m>
	v46	Interface distance from floor <m>
	v47	Interface orientation <Degrees from North>
	v48	Interface distance from room center <m>
	v49	Distance between centers of adjoining spaces <m>
Material Properties	v50	Interface frame material <Wood, stone, concrete, drywall...>
	v51	Interface infill material <Wood, stone, concrete, drywall, none...>

Table 2 .3: Variables for each network edge (By Author).

Endnotes

1.) Al-Jokhadar, Amer, and Jabi Wassim. “Humanizing The Computational Design Process.” In Parametricism VS. Materialism Evolution of Digital Technologies for Development. (London: Imperial House Publishers, 2016), p.2.

2.) Ibid. p.4.

3.) Ibid. p.7.

4.) Merrell, Paul, Eric Schkufza, and Valdlen Koltun. “Computer-Generated Residential Building Layouts.” ACM Transactions on Graphics, December 2010. p.3. https://doi.org/10.1145/1866158.1866203.

5.) Ibid. p.4.

6.) Ibid. p.5-6.

7.) Ibid. p.6.

8.) Ibid. p.6-7.

9.) As, Imdat, Siddharth Pal, and Prithwish Basu. “Artificial Intelligence in Architecture: Generating Conceptual Design Via Deep Learning.” (International Journal of Architectural Computing 16, 2018), p.311.

10.) Ibid.

11.) Ibid. p.318-319.

12.) Ibid. p.319-322.

TYPOLOGY

Because of ease of access and because it is possible to find a rich set of data within them, I have chosen the United Church as a building typology to continue this research into practice. Dominion-Chalmers United Church in Ottawa is a large scale united church usable as a case study into united churches in general. It contains many of the program elements that can be found in most smaller scale churches allowing it to be applicable to smaller case studies analyzed later.

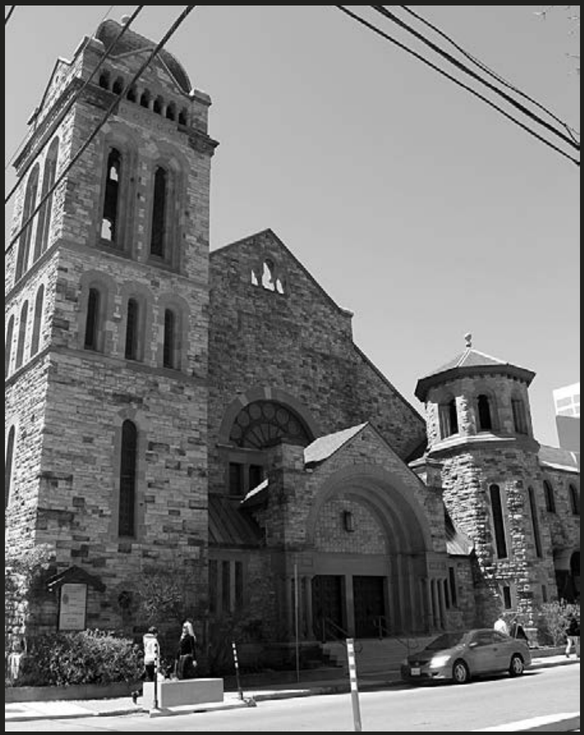
The church is a byzantine revivalist design which has a sanctuary with a square plan and a domed roof attached to a narthex with two towers on its front facade.[1] This is atypical of many smaller American protestant churches which opt for a rectilinear barn style sanctuary. However, you can see that this is a modern protestant design from how much of the program is dedicated to tasks beyond worship. There is a library, multipurpose classroom spaces, nursery, church hall large kitchen, office wing, chapel, parlour, and a manse. The complex as a whole is designed to function not only as a worship space, but also as a community center where meetings, events, and educational programs can be held.

This church complex houses most of the functions that you could find in a typical United Church with dedicated rooms for many of them. However, the affordances which you will find in this building can be found in any United Church, they just may not have a dedicated room for it. The core of a United Church is the multipurpose room that functions as a community gathering space. Functions that are a priority for each congregation are visible through which activities have a dedicated space and which are contained within the communal spaces. Looking at Dominion Chalmers and its defined programs, we can take inventory of these program elements in order to find a list of general active affordances.

Office Space – A Church can contain a few different types of office space. Administrative office space is where the church secretary works and keeps financial records for the building. The Minister’s office is where the minister can be found during their working hours if someone needs to get in contact with them, or they are working on their sermon or other administrative or pastoral care tasks. In some cases, other ancillary staff could have office space as well, like a youth group leader, or the custodian. A Space like this has to afford sitting for long periods of time, working on paperwork or at a computer, making documents, meeting with a few people at a time; and storing files, books, and office supplies.

Multipurpose Space - This space would be more commonly known as a conference space. It could be used most often as a Sunday school classroom, or a conference room for church business. It is also a room that can be used by a community group for some kind of meeting or seminar. Because of this it needs to afford the ability to sit for short periods of time, to meet in a small group, to do paperwork on some kind of surface, to present to a small group, to teach to a small group, and to store craft supplies and items used by a community group.

Maintenance - The maintenance room is a storage space where the custodian keeps their cleaning supplies. Sometimes it also functions as a space where custodial items such as keys to the building and other securi-



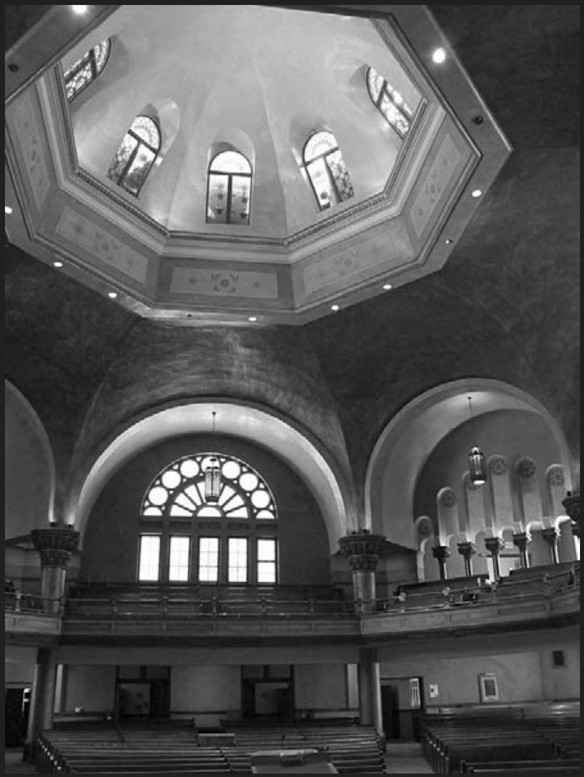


Figure 3.1: Looking North on O’Connor Street (Rathwell, 2017).

Figure 3.2: View of the Dome From Behind the Pulpit (Rathwell, 2017).

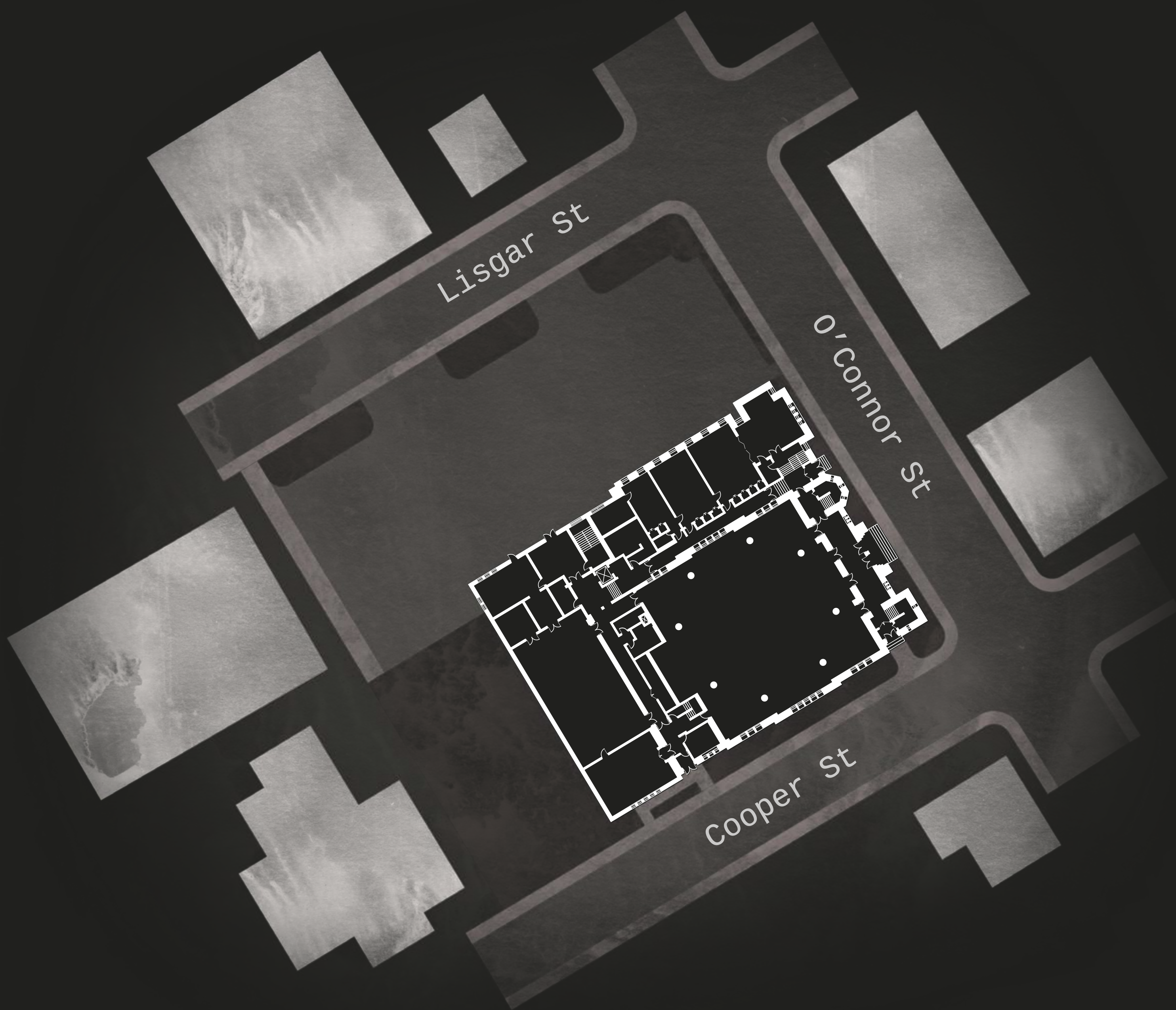


Figure 3.3: Dominion-Chalmers Site Plan (By Author).

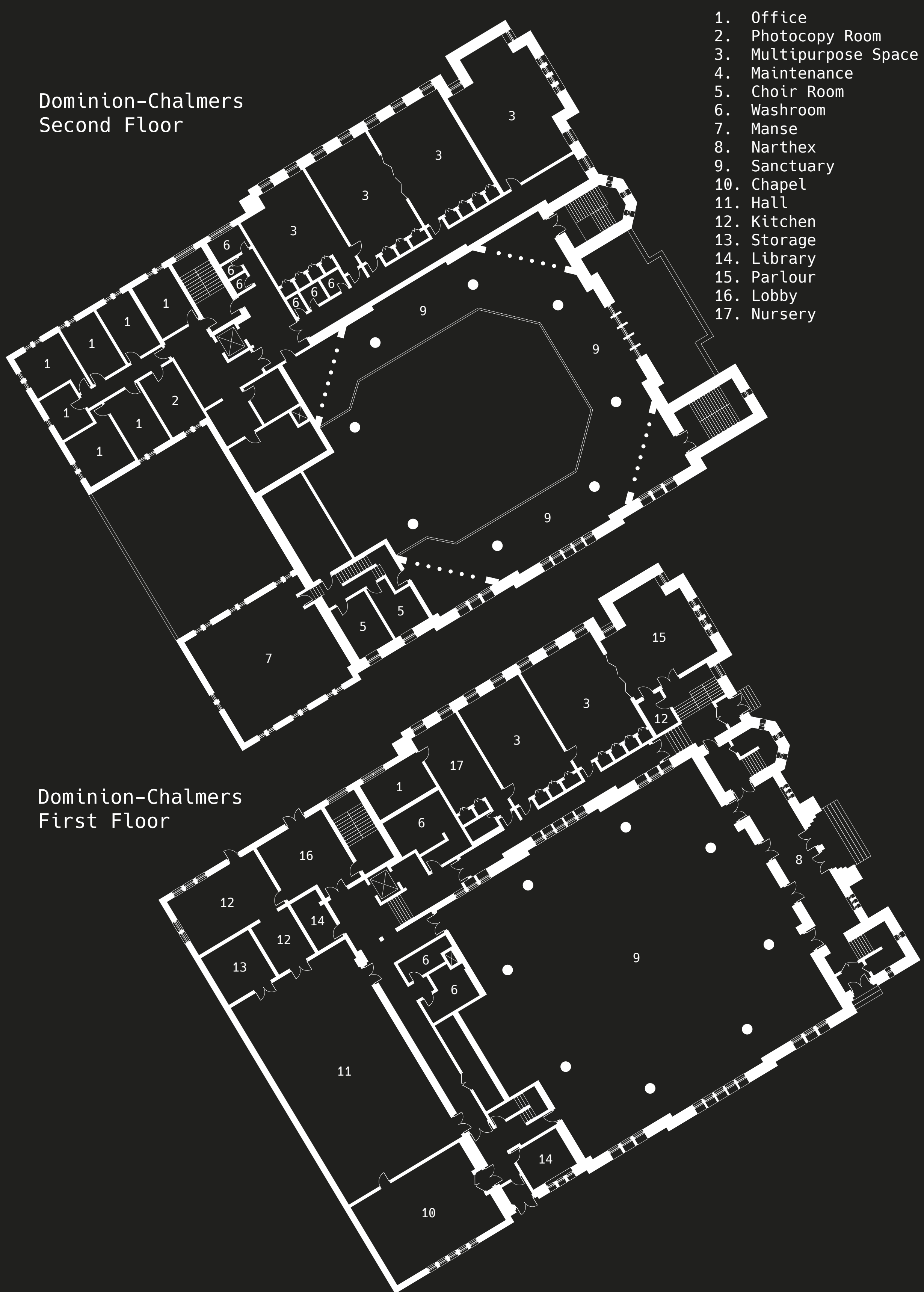


Figure 3.4 : Dominion-Chalmers Floor Plans (Reconstructed By Author from Rathwell, 2017).

ty objects are stored. It simply needs to afford the storage of these cleaning supplies and keys.

Choir Room - The choir room is the room where a church choir meets before the service to gather their things. It contains their vestments, instruments, and music. Sometimes it will also contain a piano so they can practice before the service. It needs to afford the storage of those items and the ability to play an instrument to warm up.

Manse - The manse is a residential structure built to house the minster and their family. Often they were built as separate structures from the church themselves. Today they are mostly outmoded structures. Ministers prefer to have the privacy of a residence that isn't known to everyone in the congregation, and today it is often one of the first assets which is sold to cover church costs. As a result very few churches still have a manse. In the case of Dominion-Chalmers, the manse was built into the complex itself.

Narthex - The narthex is a vestibule at the front of a church which acts as a transition space between the outdoors and the sanctuary. Sometimes it also acts as a cloakroom to store garments. It affords the ability to greet one another for the first time in a while, sit, wait, and talk before the service, and store people's vestments when they come in off of the street.

Sanctuary - The sanctuary is the central chamber in a church where people meet for worship. It often takes the form of an auditorium filled with pews or other form of seat facing towards a chancel or dais. It affords the ability to meet in a large group, to sit for worship, to pray, and to store hymn books, bibles, and ritual items. It will also contain some form of communion table, baptismal font, pulpit, and musical instrument, to afford the ability to perform baptism, communion, a sermon, or music.

Chapel - The chapel is a space of silent meditation and personal prayer. Church services are not normally held here. It affords the ability to meet in a small group, sit for worship, pray, and store bibles.

Hall - The hall is the heart of the church community. The majority of events are held here and it is the place that people meet with one another after the service. It is the most varied and versatile space in the church building needing to accommodate many different program elements. It affords the ability to meet in a large group, sit short term, serve food, eat and drink, perform music and theatre, give presentations, and store chairs and tables.

Kitchen - The kitchen is almost always adjoined to the hall space to support its function. It affords the ability to cook, clean dishes, and store kitchen tools and food.

Affordances of Space in a United Church	
Root Action	Extra Context
Sitting	Long Term
	Short Term
	For Worship
Meeting	With a Few People
	in a Small Group
	in a Large Group
Working	With Paperwork
	at a Computer
Storing	Office Supplies
	Files
	Books
	Craft Supplies
	Community Items
	Cleaning Tools
	Custodial Items
	Clothing
	Instruments
	Decorations
	Sheet Music
	Ritual Items
	Chairs and Tables
	Kitchen Tools
	Food
Practicing	Music
Performing	Music
	Theatre
	a Baptism
	Communion
	a Sermon
Making	Documents
	Crafts
Cleaning	Dishes
Watching	Children
Teaching	---
Eating	---
Drinking	---
Serving	---
Cooking	---
Greeting	---
Waiting	---
Praying	---
Playing	---

Table 3.1: United Church Affordances of Space (By Author).

Library - The library is a space which holds theological texts and literature which are shared by the community. It affords a space to sit and read as well as store books.

Parlour - The parlour is the smaller version of the main hall. It's used to host more intimate gatherings with more comfortable furniture. Quite often it also has a kitchenette attached to it so food can be served here as well. It affords the ability to meet in a small group, sit short term, eat and drink, and serve food.

Nursery - The nursery is a place for small children to gather and be watched by their parents, or another child care provider. It may also have a Sunday school for older children to be taught and entertained. It affords the ability to watch and care for children, for children to play, and for adults to teach and sit short term.

Using this case study as an inventory I have created a table of active affordances of space in United Churches (Table 3.1) Every United Church should satisfy these building affordances in one way or another and this list can be used as a key to categorize the types of space within these buildings.

Site Selection And Analysis

There are 14 currently active United Churches in Greater Sudbury. As a part of the site selection process members of various Sudbury church communities were consulted to see what their sense was about possible development sites. Demographic information about finances, congregation sizes, and physical proximities were also consulted to come to a decision about which set of these congregations would be the most rewarding to focus in on for the final design application.

For reasons of practicality, it made a lot of sense to use the sites where United Churches already existed. They are pieces of land which already have United Church history on them, are already known to the public as church property, and most importantly are already owned by the communities in question. Money is an important factor to the sustainability of these communities, so unnecessary costs like purchasing new land should probably be avoided. This immediately narrows down the possible sites to the existing 14 United Churches (Figure 3.5).

The next consideration taken into account for this project is the scale of the buildings chosen for the proposed design. The scale of a proposed design should be proportional to the size of the combined congregations. Combining congregations is a decision that would be based mostly on proximity, and need.

In 1972 the communities of Dowling, Levack, and Onaping were merged into the municipality of Onaping Falls by the provincial government. In 2001 the city of Greater Sudbury was formed, and amid anti-amalgamation sentiment in 2005 the current 12 ward system of Greater Sudbury lumped Chelmsford in with Onaping Falls under Ward 3.[2]

When talking about the urban design of Greater Sudbury the focus is usually on the downtown core, and about recentering the city to reduce the urban sprawl and car culture that has made the city unwalkable. However we often forget about the satellite communities in Greater Sudbury, which are not a result of this culture. They are thriving communities with rich histories. Most of these communities harbor a lot of animosity towards the municipality for taking their taxes and then failing to provide the services that they were originally able to provide for themselves.

Levack, Dowling, Onaping have been figuring out how to be in community together for about 50 years, and under Greater Sudbury, Chelmsford has joined them in a single municipal region. (Figure 3.6) There is an opportunity here to strengthen the community within Ward 3 and bring the townships of Levack, Dowling Onaping and Chelmsford together with a new community center that honors their rich histories both separately and together.

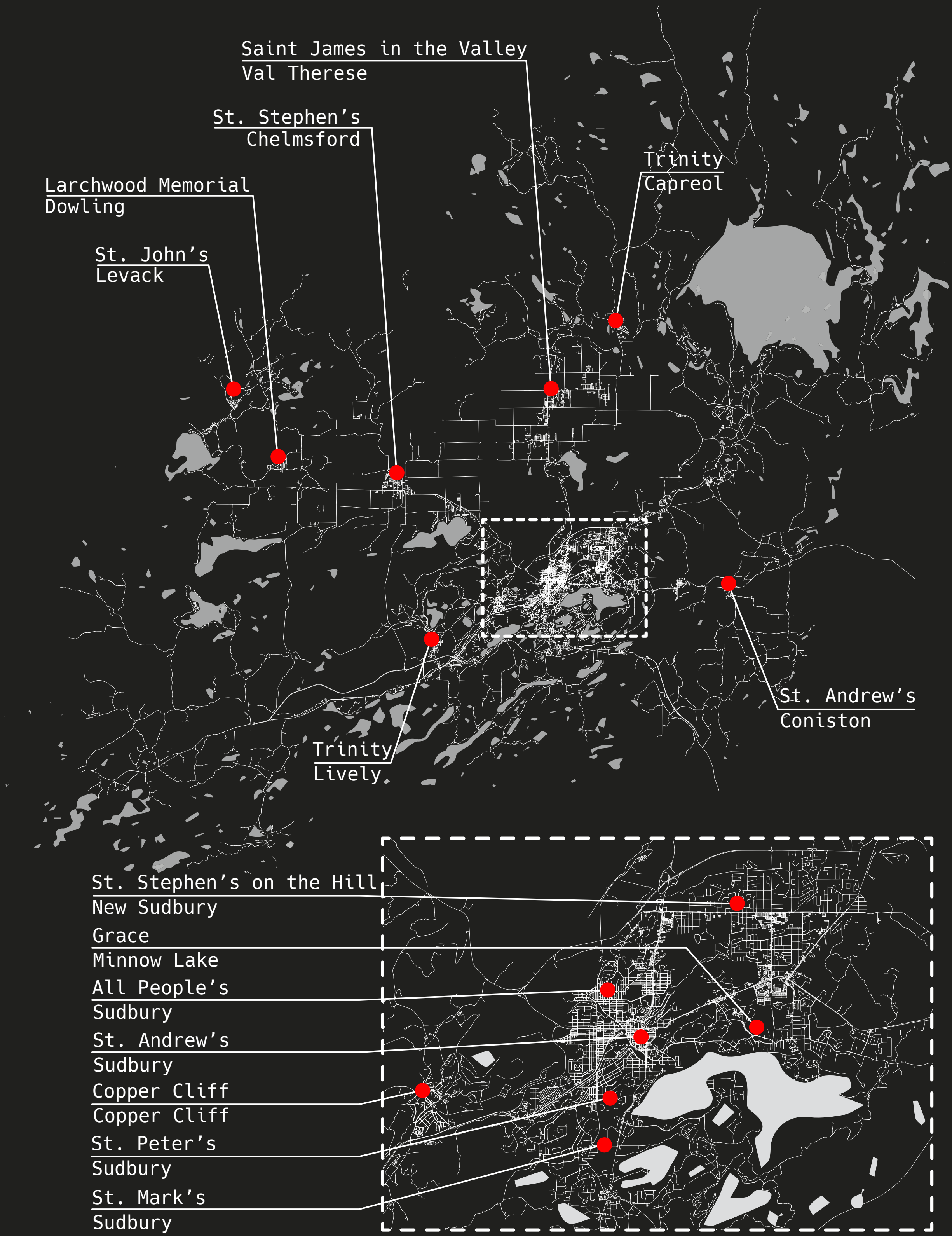


Figure 3.5: 14 United Churches in Greater Sudbury (By Author).

Greater Sudbury Ward 3

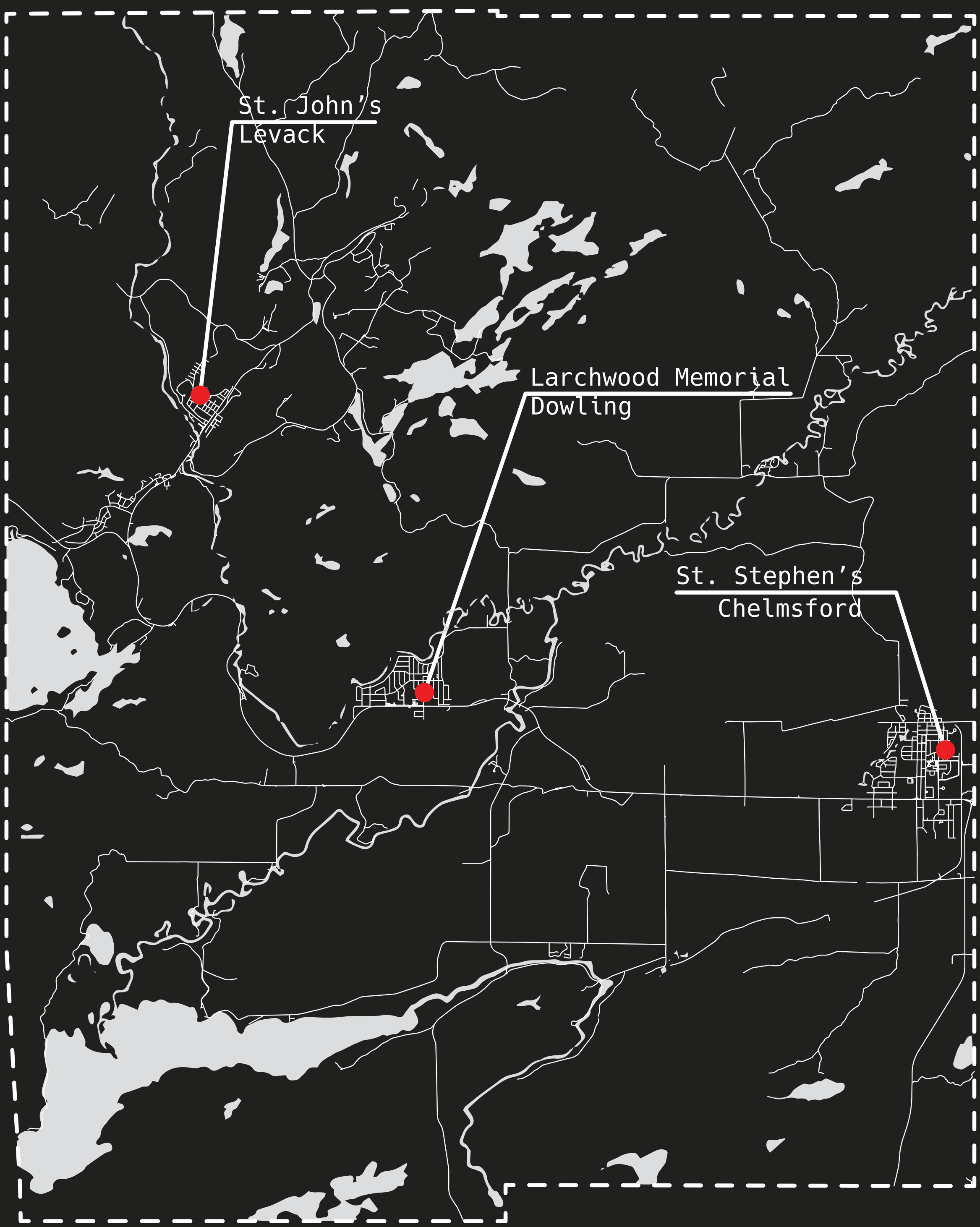


Figure 3.6: Greater Sudbury Ward 3 (By Author).

St. Stephen’s in Chelmsford has been looking at closing down for some time now. If that were to happen, the remaining congregation would do what they do every summer when they close to cut costs, and travel to Larchwood Memorial for their services. St. John’s and Larchwood Memorial are part of the same pastoral charge, which means that they already share the costs for a minister and the maintenance of their buildings.

All of these churches are old buildings built in the 50s and 60s, and the congregations there are small enough that there hasn’t really been any accessibility retrofitting. If they pooled their resources they could build a new church building that would be able to meet their needs now and into the future.

Due to being central to all of these communities, the ideal location would be at the current site of Larchwood Memorial United Church in Dowling. Dowling is an outlier among the constellation communities of Greater Sudbury. It originally started as a farming village and today it is the largest community in Onaping falls, and seen as the hub of the area with many different amenities.

The site of Larchwood Memorial United Church is located close to the main highway on a triple sized lot. (Figure 3.7) It is a small barn style church with a two story wing on the west side of the building dedicated to non-worship activities. It was constructed under the labor and direction of the congregation in 1965. The general proposal is to replace the existing building with a new building on it’s current location keeping all the program elements on a single level to improve accessibility.

St. Stephen’s in Chelmsford, Larchwood Memorial in Dowling, and St John’s United Church in Onaping, will also be used as case studies to collect data about united church buildings of the size and scale required to hold an amalgamation of these three congregations. This data will be used during the generative process and can be found in the remainder of this chapter.

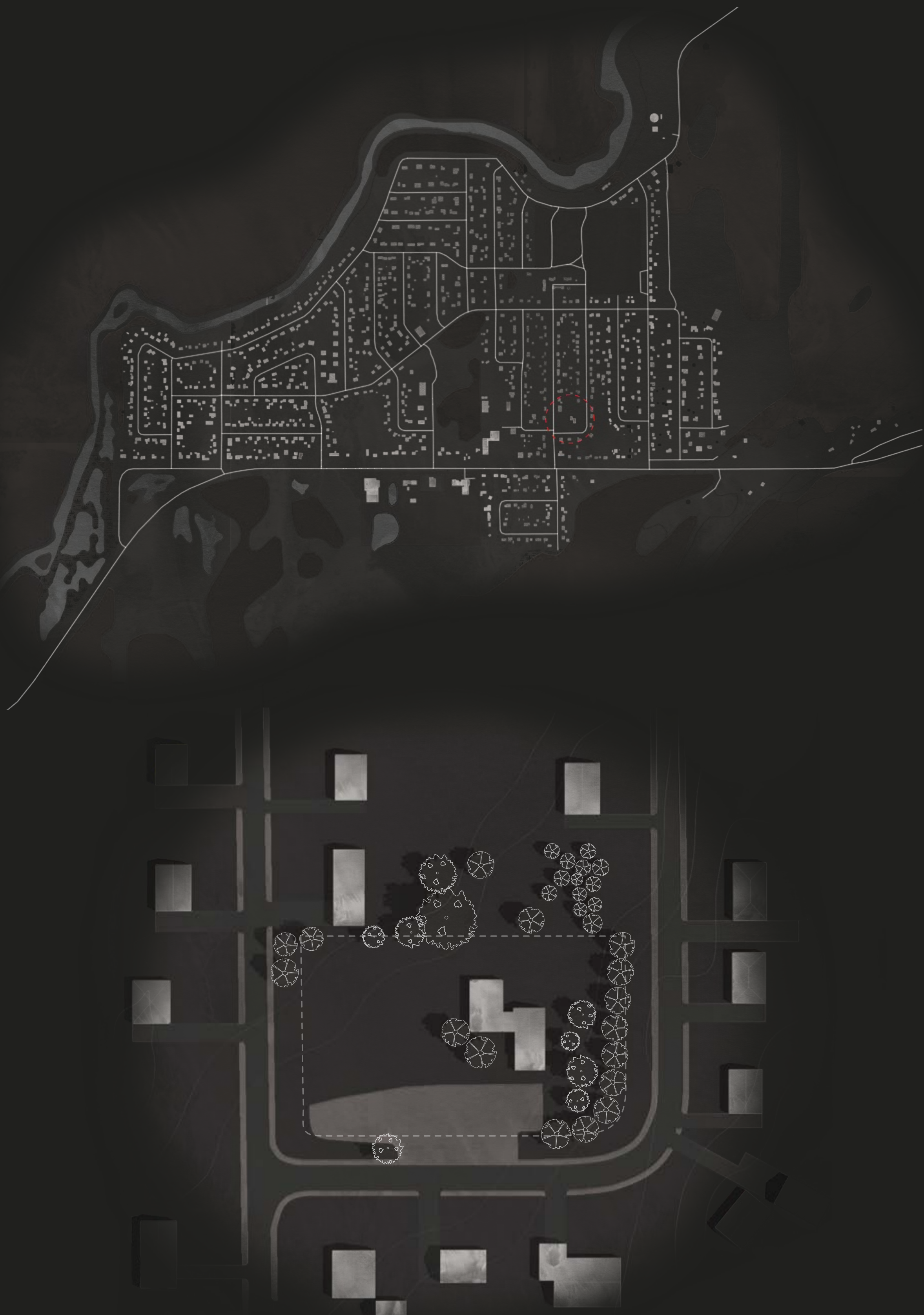


Figure 3.7: Dowling and Larchwood Memorial United Church (By Author).

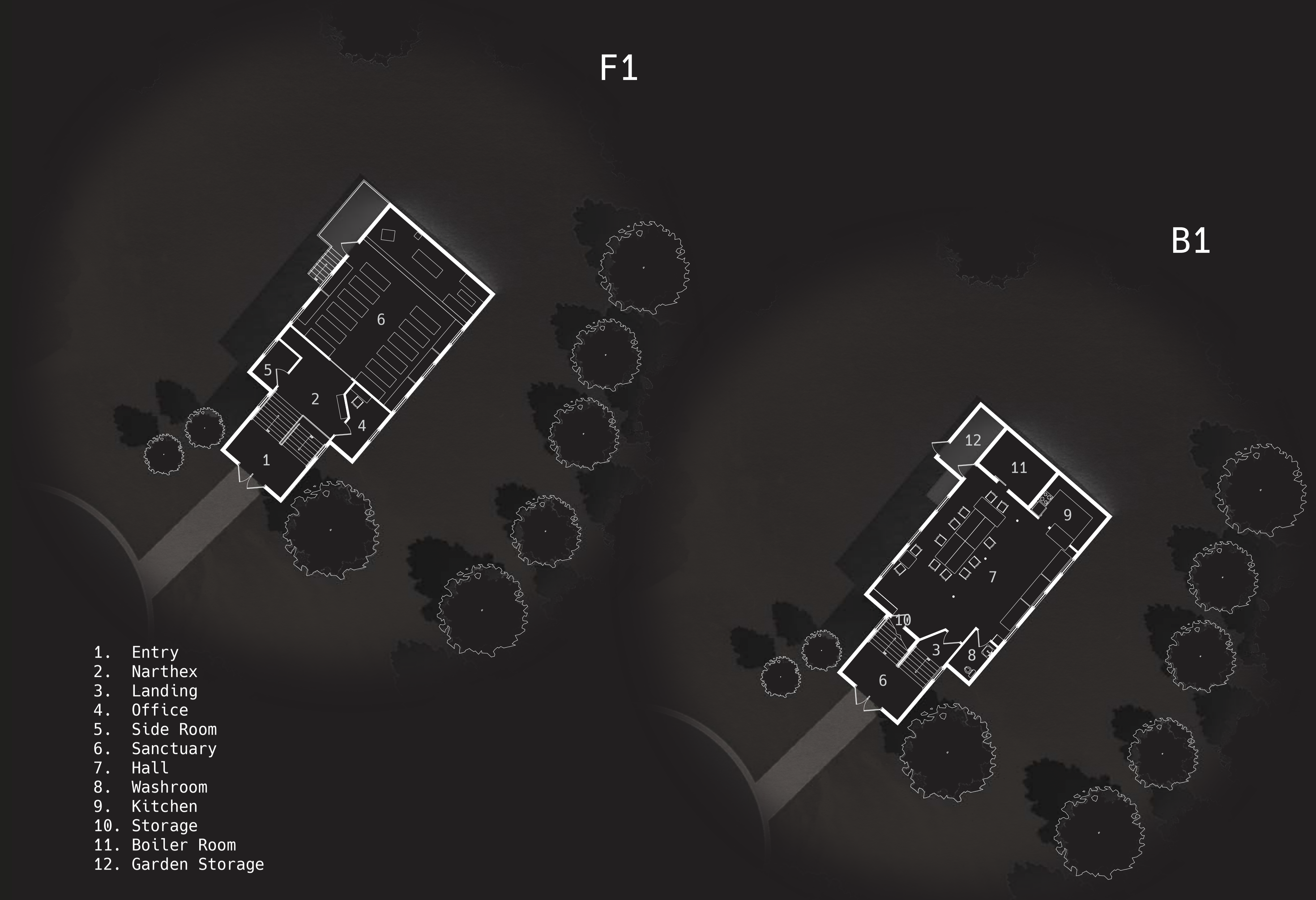
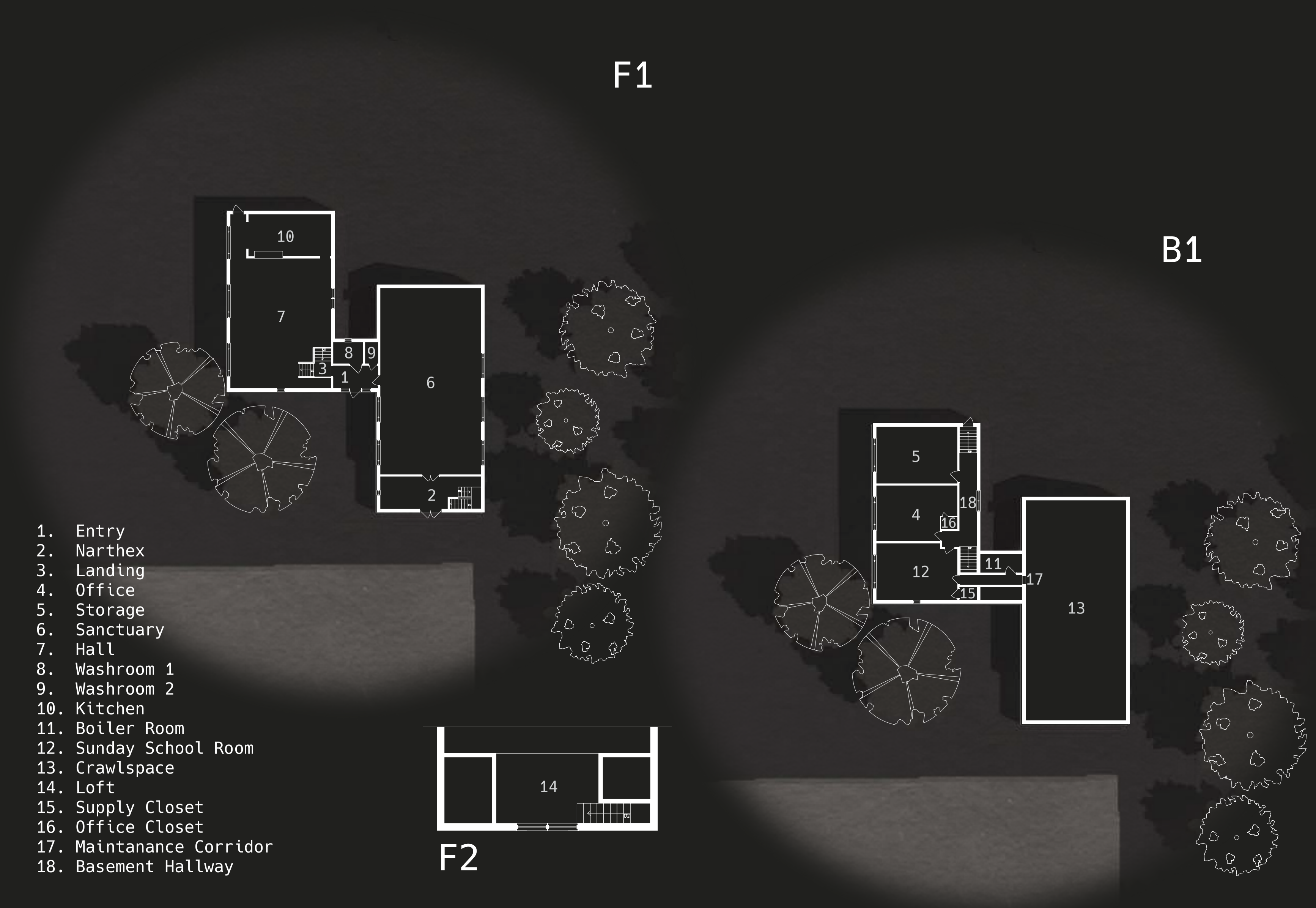


Figure 3.8: St. Stephen’s United Church Plans (By Author).

		Space Width	Space Length	Space Height	Space Area	Aspect Ratio	Relative Illumunance	Background Noise	Average Temperature	Floor Material	Wall Material	Ceiling Material
1.	Entry	2400mm	4267mm	4165mm	10.2m^2	0.5624	0.1074	25db	17° C	Vinyl Tile	Drywall	Drywall
2.	Narthex	3454mm	4627mm	3276mm	14.7m^2	0.7465	0.1039	26db	17° C	Vinyl Tile	Drywall	Drywall
3.	Landing	1829mm	1981mm	2260mm	3.1m^2	0.9233	0.05366	22db	17° C	Vinyl Tile	Drywall	Drywall
4.	Office	2692mm	3454mm	2641mm	7.0m^2	0.7794	0.1779	26db	17° C	Vinyl Tile	Drywall	Drywall
5.	Side Room	1612mm	2425mm	2541mm	3.9m^2	0.6647	0.2254	26db	17° C	Vinyl Tile	Drywall	Drywall
6.	Sanctuary	7721mm	8915mm	2870mm	68.8m^2	0.8661	0.09213	26db	17° C	Carpet	Panelling	Ceilling Tile
7.	Hall	7721mm	9740mm	2260mm	68.2m^2	0.7927	0.03754	22db	17° C	Carpet	Vinyl	Panelling
8.	Washroom	1613mm	2540mm	2260mm	4.1m^2	0.6350	0.0	25db	17° C	Vinyl	Drywall	Drywall
9.	Kitchen	2629mm	3734mm	2260mm	9.8m^2	0.7041	0.001072	22db	18° C	Vinyl	Panelling	Panelling
10.	Storage	1828mm	2083mm	1524mm	3.8m^2	0.8776	0.0	22db	17° C	Concrete	Drywall	Drywall
11.	Boiler Room	2540mm	3785mm	2438mm	9.6m^2	0.6711	0.0	28db	18° C	Concrete	Concrete	Concrete
12.	Garden Storage	1815mm	3811mm	2438mm	6.9m^2	0.4763	0.0	22db	12° C	Concrete	Concrete	Concrete

Table 3.2: Node Data Captured from St. Stephen’s (By Author).



1. Entry
2. Narthex
3. Landing
4. Office
5. Storage
6. Sanctuary
7. Hall
8. Washroom 1
9. Washroom 2
10. Kitchen
11. Boiler Room
12. Sunday School Room
13. Crawlspace
14. Loft
15. Supply Closet
16. Office Closet
17. Maintanance Corridor
18. Basement Hallway

Figure 3.9: Larchwood Memorial United Church Plans (By Author).

		Space Width	Space Length	Space Height	Space Area	Aspect Ratio	Relative Illumunance	Background Noise	Average Temperature	Floor Material	Wall Material	Ceiling Material
1.	Entry	1575mm	3149mm	2413mm	4.9m^2	0.5002	0.06777	27db	17° C	Vinyl Tile	Drywall	Drywall
2.	Narthex	2260mm	7061mm	2108mm	12.9m^2	0.3201	0.08353	26db	17° C	Carpet	Drywall	Ceilling Tile
3.	Landing	1117mm	1219mm	4060mm	1.3m^2	0.9163	0.48146	27db	17° C	Vinyl Tile	Panelling	Ceilling Tile
4.	Office	3912mm	5689mm	2413mm	20.1m^2	0.6876	0.12843	24db	17° C	Carpet	Panelling	Ceilling Tile
5.	Storage	3988mm	5689mm	2413mm	22.7m^2	0.7010	0.12686	24db	17° C	Carpet	Panelling	Ceilling Tile
6.	Sanctuary	7061mm	13055mm	4521mm	92.2m^2	0.5409	0.03275	27db	17° C	Vinyl Tile	Drywall	Ceilling Tile
7.	Hall	7061mm	8780mm	2692mm	63.8m^2	0.8042	0.48146	27db	17° C	Vinyl Tile	Panelling	Ceilling Tile
8.	Washroom 1	1498mm	2082mm	2413mm	3.1m^2	0.7195	0.94001	27db	17° C	Vinyl Tile	Drywall	Drywall
9.	Washroom 2	914mm	1498mm	2413mm	1.3m^2	0.6101	0.0	27db	18° C	Vinyl Tile	Drywall	Drywall
10.	Kitchen	2946mm	5791mm	2692mm	17.0m^2	0.5087	0.11618	29db	17° C	Vinyl Tile	Panelling	Ceilling Tile
11.	Boiler Room	1295mm	2946mm	1930mm	3.8m^2	0.4396	0.0	29db	18° C	Concrete	Concrete	None
12.	Sunday School Room	3988mm	5689mm	2413mm	22.2m^2	0.7010	0.19211	24db	12° C	Carpet	Panelling	Ceilling Tile
13.	Crawlspace	7061mm	15468mm	1930mm	109.2m^2	0.4565	0.0	29db	12° C	Concrete	Concrete	None
14.	Loft	2413mm	3505mm	2209mm	8.4m^2	0.6884	0.10897	29db	17° C	Carpet	Drywall	Ceilling Tile
15.	Supply Closet	914mm	1219mm	2413mm	1.1m^2	0.7498	0.0	24db	17° C	Carpet	Panelling	Ceilling Tile
16.	Office Closet	838mm	1143mm	2413mm	0.9m^2	0.7332	0.0	24db	17° C	Carpet	Panelling	Ceilling Tile
17.	Maintanance Corridor	711mm	4419mm	1930mm	3.1m^2	0.1609	0.0	29db	11° C	Concrete	Concrete	None
18.	Basement Hallway	1219mm	7823mm	2413mm	9.5m^2	0.1558	0.06888	24db	12° C	Carpet	Panelling	Ceilling Tile

Table 3.3: Node Data Captured From Larchwood Memorial (By Author).

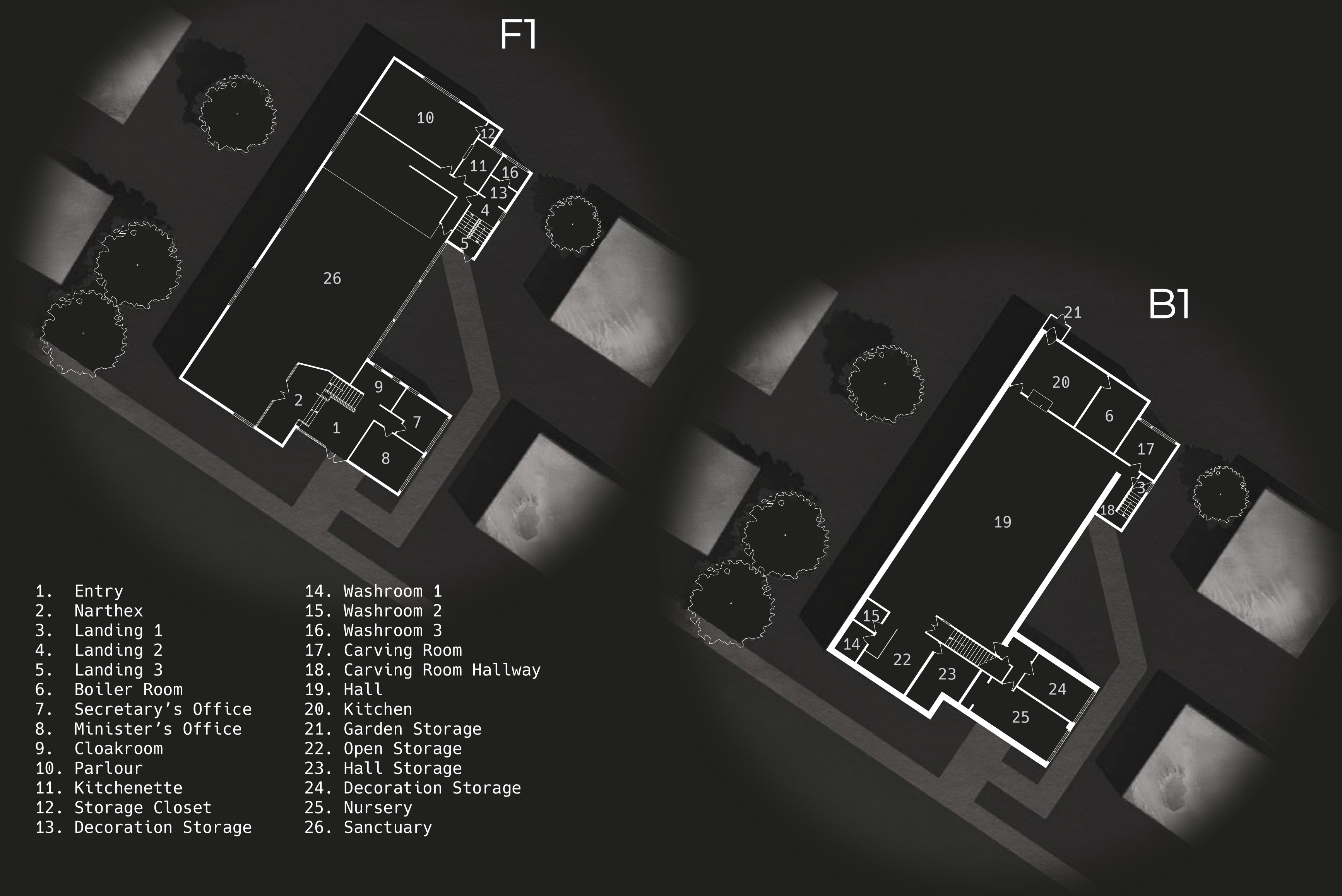


Figure 3.10: St. John’s United Church Plans (By Author).

		Space Width	Space Length	Space Height	Space Area	Aspect Ratio	Relative Illumunance	Background Noise	Average Temperature	Floor Material	Wall Material	Ceiling Material
1.	Entry	4140mm	4534mm	3556mm	18.7m^2	0.9131	0.7924	27db	17° C	Carpet	Drywall	Ceilling Tile
2..	Narthex	2184mm	6146mm	3023mm	12.7m^2	0.3554	0.1009	26db	18° C	Carpet	Drywall	Ceilling Tile
3.	Landing 1	1295mm	1549mm	3378mm	2.0m^2	0.8360	0.0	27db	17° C	Concrete	Drywall	Ceilling Tile
4.	Landing 2	1422mm	2209mm	2286mm	3.1m^2	0.6437	0.1234	24db	17° C	Vinyl	Drywall	Drywall
5.	Landing 3	990mm	2209mm	2337mm	2.2m^2	0.4482	0.4555	24db	17° C	Vinyl	Drywall	Drywall
6.	Boiler Room	3264mm	2394mm	3378mm	14.3m^2	0.7428	0.0	27db	18° C	Vinyl	Concrete	Ceilling Tile
7.	Secretary's Office	3137mm	3555mm	2743mm	9.8m^2	0.8824	0.1671	27db	16° C	Carpet	Drywall	Drywall
8.	Minister's Office	3289mm	4203mm	2794mm	13.8m^2	0.7825	0.0631	27db	17° C	Carpet	Drywall	Drywall
9.	Cloakroom	1981mm	3365mm	3048mm	6.6m^2	0.5887	0.2670	27db	18° C	Carpet	Drywall	Drywall
10.	Parlour	4533mm	7962mm	3023mm	36.1m^2	0.5693	0.2798	29db	17° C	Carpet	Drywall	Drywall
11.	Kitchenette	2108mm	3111mm	2997mm	6.5m^2	0.6776	0.2615	29db	18° C	Vinyl	Drywall	Drywall
12.	Storage Closet	953mm	1308mm	3023mm	1.2m^2	0.7286	0.0	24db	12° C	Carpet	Drywall	Drywall
13.	Decoration Storage	1460mm	2209mm	2438mm	3.2m^2	0.6609	0.1414	29db	12° C	Vinyl	Drywall	Drywall
14.	Washroom 1	1701mm	1752mm	2438mm	2.9m^2	0.9709	0.0	29db	17° C	Vinyl	Tile	Ceilling Tile
15.	Washroom 2	1752mm	2387mm	2438mm	4.1m^2	0.7340	0.0	24db	18° C	Vinyl	Tile	Ceilling Tile
16.	Washroom 3	1536mm	2209mm	3023mm	3.4m^2	0.6953	0.1783	24db	17° C	Vinyl	Drywall	Drywall
17.	Carving Room	2971mm	3326mm	3327mm	9.8m^2	0.8933	0.3100	29db	17° C	Concrete	Drywall	Drywall
18.	Carving Room Hallway	1060mm	4038mm	2311mm	4.2m^2	0.2625	0.0	25db	17° C	Concrete	Drywall	Drywall
19.	Hall	9728mm	18110mm	2489mm	167.5m^2	0.5372	0.0	27db	17° C	Concrete	Panelling	Ceilling Tile
20.	Kitchen	4394mm	5384mm	3658mm	23.7m^2	0.8161	0.0	27db	18° C	Vinyl	Panelling	Ceilling Tile
21.	Garden Storage	1117mm	1574mm	3099mm	1.7m^2	0.7097	0.0	27db	17° C	Concrete	None	None
22.	Open Storage	4025mm	4394mm	2489mm	17.7m^2	0.9160	0.0	27db	17° C	Concrete	Panelling	Ceilling Tile
23.	Hall Storage	3352mm	4508mm	2743mm	12.7m^2	0.7436	0.0	29db	17° C	Concrete	Concrete	Drywall
24.	Decoration Storage 2	2743mm	4737mm	4039mm	12.9m^2	0.5791	0.0213	26db	18° C	Vinyl	Drywall	Drywall
25.	Nursery	3200mm	6540mm	3124mm	20.1m^2	0.4893	0.0245	27db	17° C	Concrete	Drywall	Drywall
26.	Sanctuary	10185mm	22631mm	8052mm	230.5m^2	0.4500	0.1750	27db	17° C	Vinyl	Wood	Wood

Table 3.4: Node Data Captured From St. John's (By Author).

Endnotes

1.) Natalie Anderson Rathwell, “Heritage Value in Ottawa’s Domin ion-Chalmers United Church: History, Community, Sight, and Sound,” Journal of the Society for the Study of Architecture in Canada 43, no. 1 (January 10, 2018): p.49, accessed January 10, 2018, doi:https://doi.org/10.7202/1049407ar).

2.) “Community History,” The Onaping Falls News, accessed August 15, 2021, http://www.onapingfallsnews.com/community-history/)

Application Of The Generative System

This section showcases the use of the collected data for generating schematic designs using the generative algorithm. Starting program elements were chosen randomly based upon their likelihood of being included in a building from the original dataset. This means that all essential program elements are always included, and other possible program elements are added by chance based upon adjacency to the primary elements. These examples were kept deliberately simple for ease of understanding and also because optimization time increases exponentially with complexity.

The following diagrams show what the upper and lower bounds for the cost function was, and whether or not the space within that massing model managed to bring its cost value down to zero. They follow two examples of a generated schematic design to show what the outcomes are like for this process.

You can see that for most variables, the majority of the spaces on both plans fell within the boundaries set by the collected data. In the cases where they didn’t, it could be for a number of reasons. The optimization algorithm is attempting to bring all of these cost functions down to zero together, in some cases it could be that lowering the cost of a given term causes another term to increase more than what would be gained by that action. In all cases, generative processes like these are processes of give and take which are beholden to their starting conditions. A local maximum of the fitness plane may not result in a usable design.

Area Cost

	Upper	Lower
Entry	10.24 mSq	4.95 mSq
Narthex	14.70 mSq	12.90 mSq
Hallway	9.54 mSq	9.54 mSq
Office	20.08 mSq	6.99 mSq
Sanctuary	92.18 mSq	68.83 mSq
Hall	68.23 mSq	63.80 mSq
Washroom	4.09 mSq	3.11 mSq
Kitchen	17.06 mSq	9.81 mSq
Boiler Room	9.61 mSq	3.81 mSq
Storage	22.68 mSq	3.80 mSq
Garden Storage	6.92 mSq	6.92 mSq

	Plan A Cost	Plan B Cost
Entry	0	0
Narthex	11.5	0.1
Hallway	2.74	---
Office	0	0.49
Sanctuary	0	10.53
Hall	0	0
Washroom	1.01	1.71
Kitchen	0	0.1
Boiler Room	0.49	0
Storage	0	0
Garden Storage	---	0

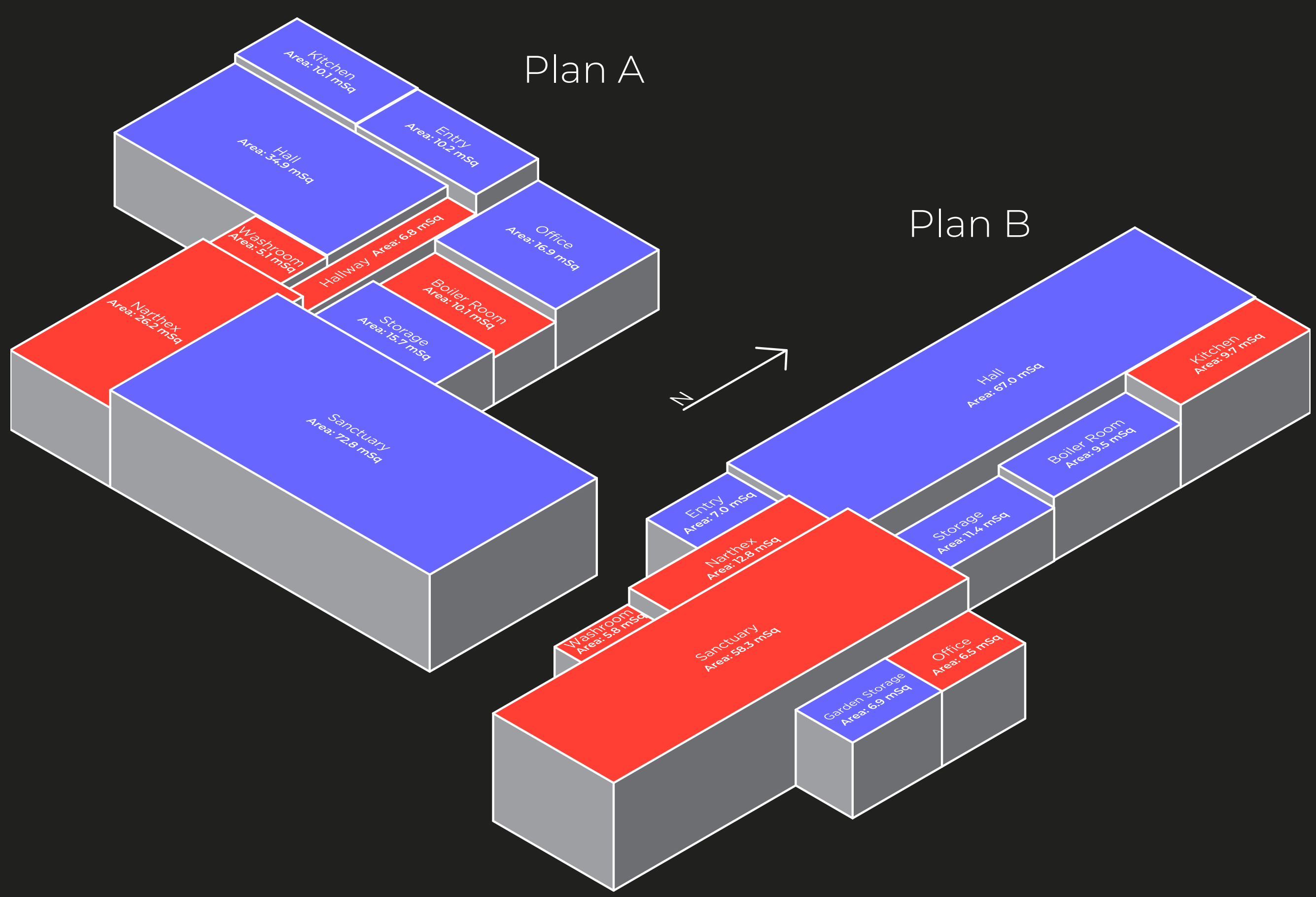
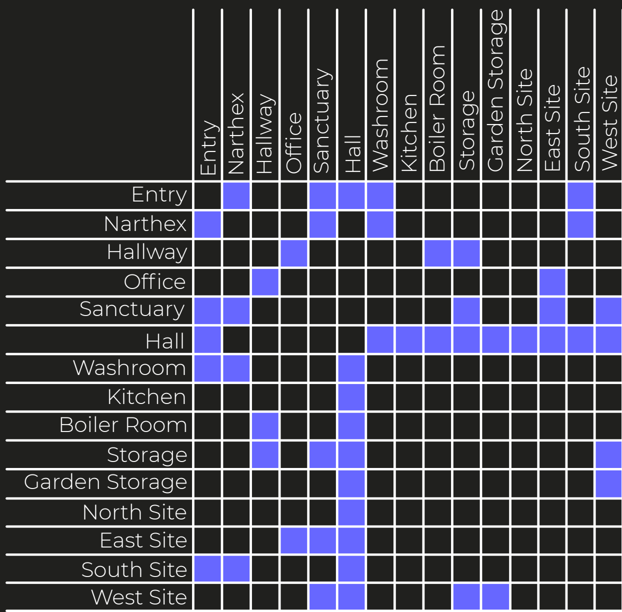


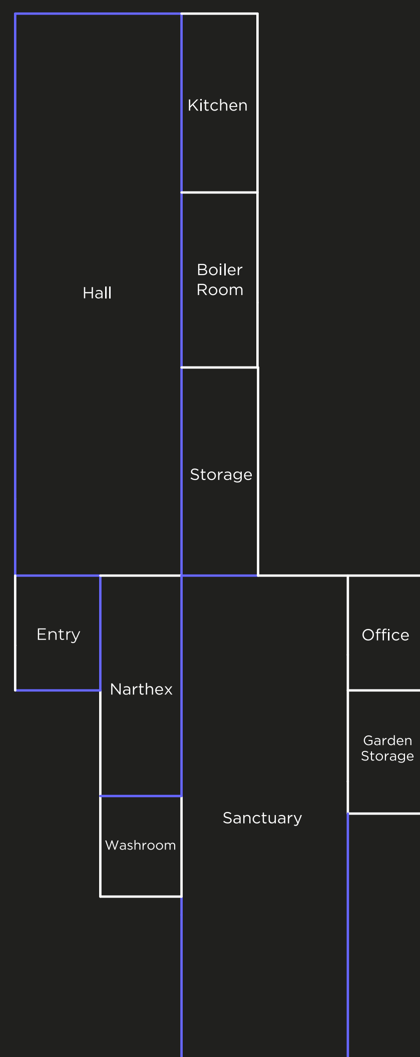
Figure 4.1: Test Cases for Optimization of Room Area (By Author).

Adjacency Cost



	Plan A Cost	Plan B Cost
Entry	5 - 1 = 4	5 - 3 = 2
Narthex	4 - 3 = 1	4 - 3 = 1
Hallway	3 - 3 = 0	---
Office	2 - 2 = 0	2 - 1 = 1
Sanctuary	5 - 3 = 2	5 - 4 = 1
Hall	10 - 5 = 5	10 - 6 = 4
Washroom	3 - 2 = 1	3 - 1 = 2
Kitchen	1 - 1 = 0	1 - 1 = 0
Boiler Room	2 - 1 = 0	2 - 1 = 0
Storage	4 - 2 = 2	4 - 2 = 2
Garden Storage	---	2 - 1 = 1

Plan B



Plan A

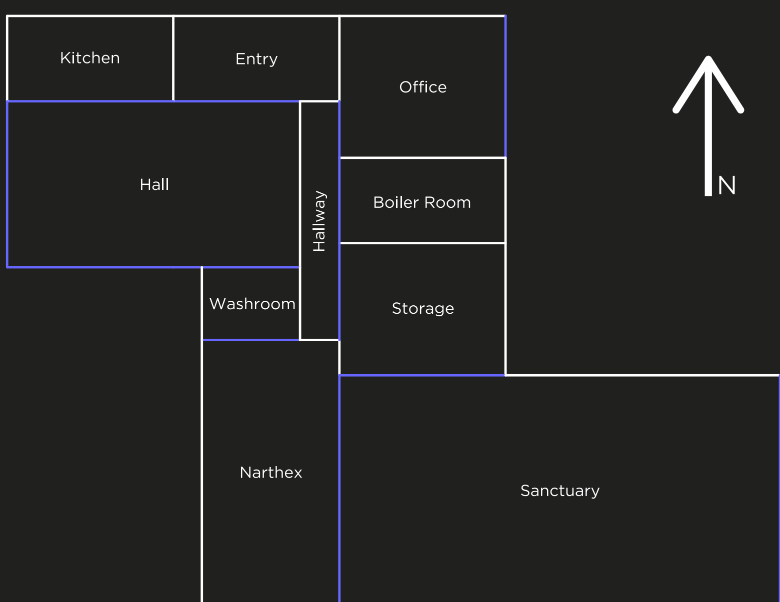


Figure 4.2: Test Cases for Optimizing Room Adjacency (By Author).

Ratio Cost

	Upper	Lower
Entry	0.562	0.500
Narthex	0.746	0.320
Hallway	0.155	0.155
Office	0.779	0.687
Sanctuary	0.866	0.540
Hall	0.804	0.792
Washroom	0.719	0.635
Kitchen	0.704	0.508
Boiler Room	0.671	0.439
Storage	0.877	0.701
Garden Storage	0.476	0.476

	Plan A Cost	Plan B Cost
Entry	0.012	0.178
Narthex	0	0
Hallway	0.009	---
Office	0.075	0.004
Sanctuary	0.018	0.201
Hall	0.225	0.497
Washroom	0.022	0.087
Kitchen	0	0.084
Boiler Room	0	0
Storage	0	0.314
Garden Storage	---	0.160

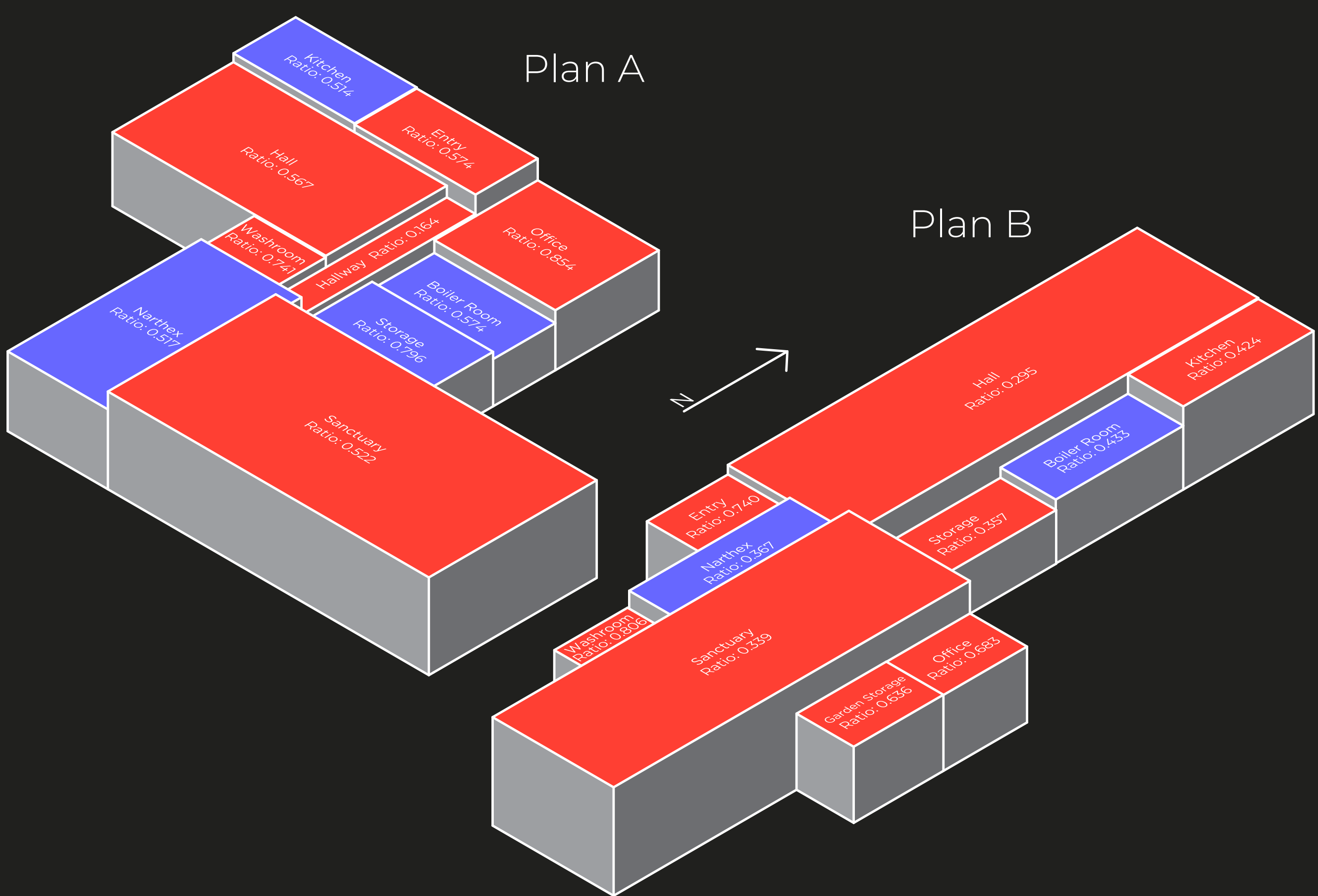


Figure 4.3: Test Cases for Optimizing the Ratio of Length to Width (By Author).

Shape Cost

	Plan A Cost	Plan B Cost
Entry	0	0
Narthex	0	0
Hallway	0	---
Office	0	0
Sanctuary	0	0
Hall	0	0
Washroom	0	0
Kitchen	0	0
Boiler Room	0	0
Storage	0	0
Garden Storage	---	0

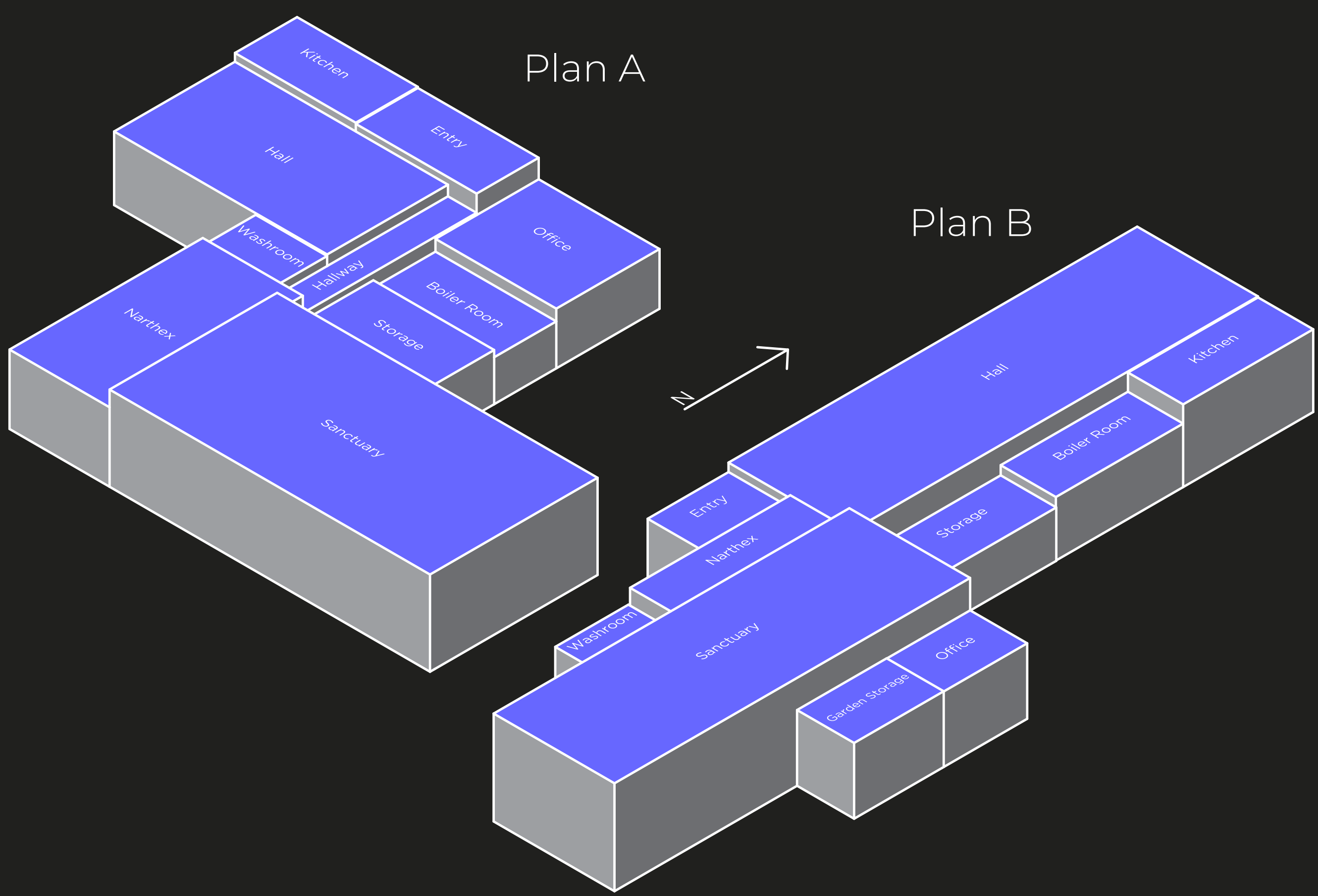


Figure 4.4: Test Cases for Optimizing Shape and Convexity of Space (By Author).

Height Cost

	Upper	Lower
Entry	4.16m	2.413m
Narthex	3.27m	2.108m
Hallway	2.41m	1.41m
Office	2.64m	2.41m
Sanctuary	4.52m	2.87m
Hall	2.69m	2.26m
Washroom	2.40m	2.26m
Kitchen	2.69m	2.26m
Boiler Room	2.43m	1.93m
Storage	2.41m	1.52m
Garden Storage	2.43m	2.43m

	Plan A Cost	Plan B Cost
Entry	0	0
Narthex	0	0
Hallway	0	---
Office	0.5	0
Sanctuary	0	0
Hall	0	0
Washroom	0	0
Kitchen	0	0
Boiler Room	0.43	0
Storage	0	0
Garden Storage	---	0

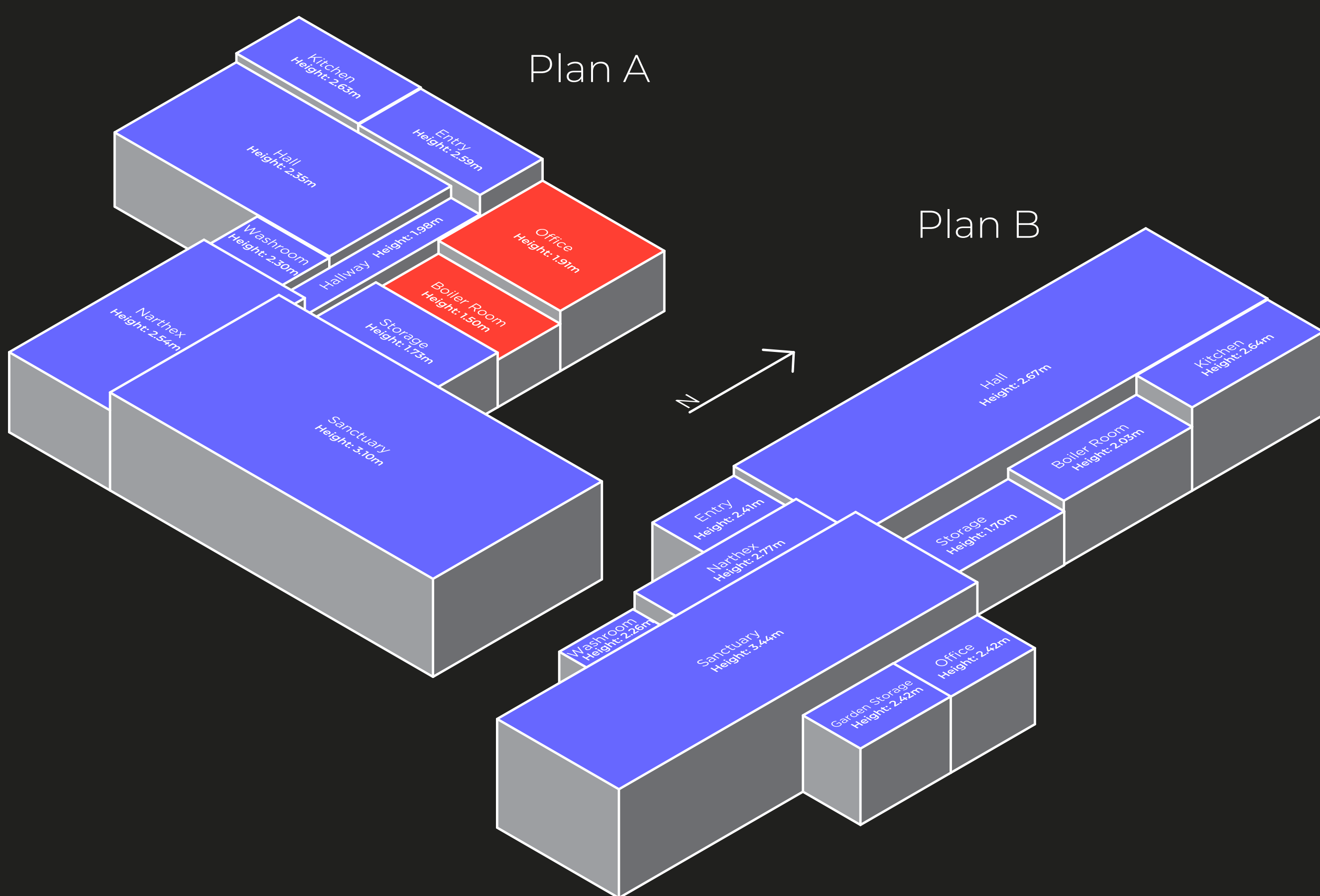


Figure 4.5: Test Cases for Optimizing Height of Space (By Author).

Solar Access Cost

	Upper	Lower
Entry	---	1.61 mSq
Narthex	---	3.58 mSq
Hallway	---	0.87 mSq
Office	---	1.11 mSq
Sanctuary	---	4.71 mSq
Hall	---	4.22 mSq
Washroom	---	0 mSq
Kitchen	---	0 mSq
Boiler Room	---	0 mSq
Storage	---	0 mSq
Garden Storage	---	0 mSq

	Plan A Cost	Plan B Cost
Entry	1.61	0
Narthex	0	0
Hallway	0	---
Office	0	0
Sanctuary	0	0
Hall	0	0
Washroom	0	0
Kitchen	0	0
Boiler Room	0	0
Storage	0	0
Garden Storage	---	0

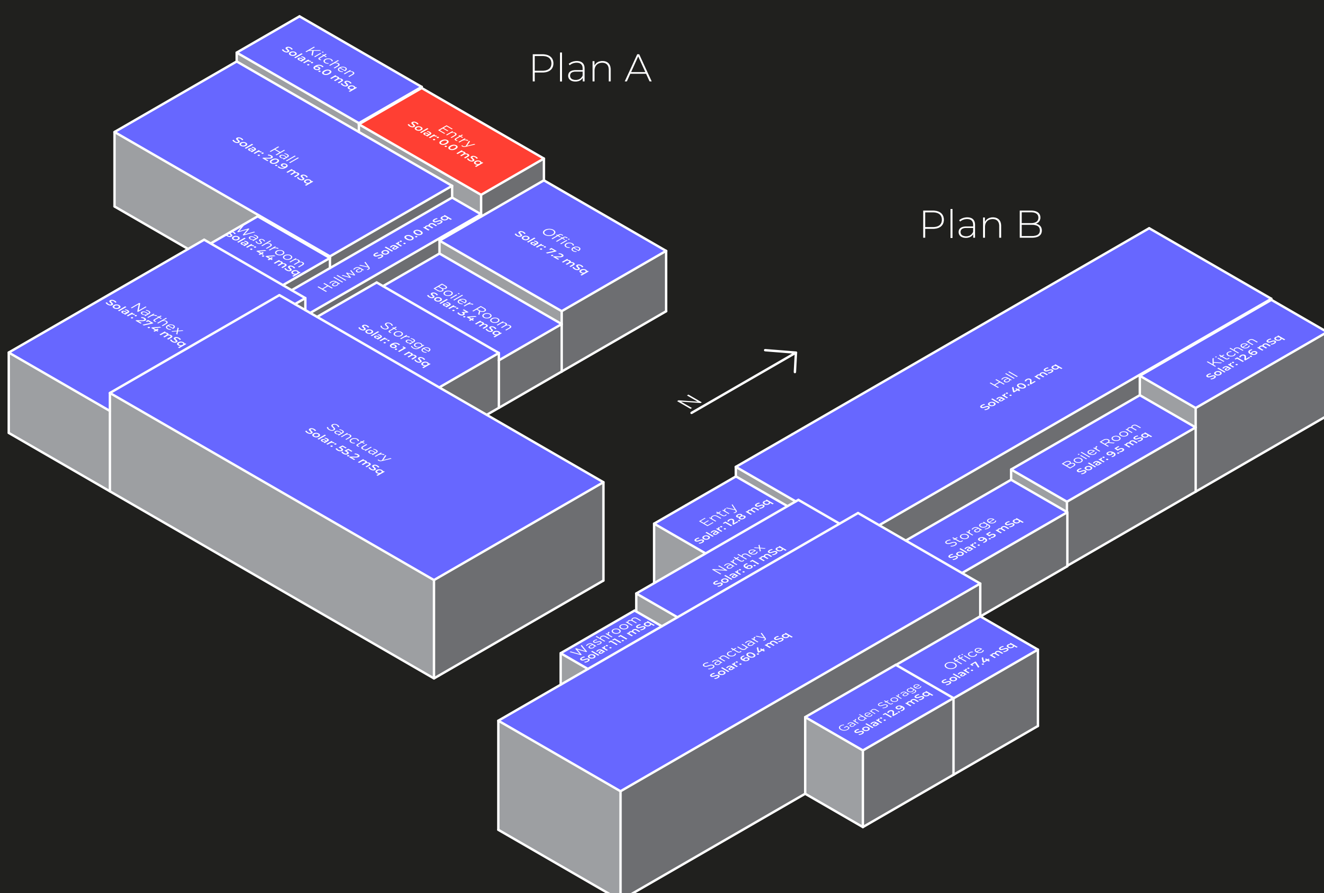


Figure 4.6: Test Cases for Optimizing Solar Access of Space (By Author).

Application Of Objects And Affordances

Each object found in the case studies which supplies an active affordance to a space was photographed and categorized by the location that it was found, and the active affordances that it provides to the space. There are forty-three different active affordances that can be attributed to a given object, and an object can be given one or more of them.

On the following page is a partial table of this information. Various objects have been inventoried and the affordances which they provide have been cataloged. (Table 5.1) Information identifying the location of the object, and the relative frequency of finding that object are also retained.

Building interfaces such as doors, windows, stairs, and other openings have also been cataloged. Information identifying their height, width, depth, the kinds of information they let pass through them, their orientation to the north, their distance from the center of the room, and their distance from the floor have all been logged in a database. (Table 5.2)

Information was also collected from the opposite perspective. For each of the spaces, there is aggregate environmental data and aggregate active affordance data to give us information about what the average room of a given type affords its environment. (Table 5.3) (Table 5.4)

This information can then be used in a process called affordance matching, where information about objects is matched to information about spaces in order to use the data to populate the spaces which were generated by the schematic algorithm. (Figure 5.1) A passthrough window affords the ability to store cleaning supplies, ritual items, and serve food, it also connects the kitchen and hall spaces. Choosing to place that object in the final design fulfills those spatial requirements for the kitchen space, and creates a physical interface between the kitchen and hall required by the type of adjacency.

Passthrough Window

Active Affordances
- Storing Cleaning Supplies
- Storing Ritual Items
- Serving Food

Context Data
- Connects Kitchen and Hall
- Height, 1016mm to 1097mm
- Width, 1473mm to 1981mm
- Depth, 101mm to 152mm
- Floor Distance, 914mm to 1041mm
- Distance From Center of Kitchen, 1464mm to 2065mm
- Distance From Center of Hall, 4074mm to 5312mm
- Frame Material, Wood
- Infill Material, None

Kitchen

Active Affordances
- Storing Cleaning Supplies
- Storing Ritual Items
- Storing Kitchen Tools
- Storing Food
- Cleaning Dishes
- Serving Food
- Cooking Food
- Phoning

Environmental Affordances
- Relative Illuminance of 0.0946
- Vinyl Tile Flooring
- Drywall Walls
- Drywall Ceiling

Figure 5.1: Affordance Mapping Process (By Author).

Table 5.1: Affordances attributed to Objects (By Author).

	Banner Hook	Baptismal Font	Book Table	Children's Toys	Child's Table	Choir Pit	Coat Rack	Communion Table	Bulletin Board	Community Showcase	Supply Shelf	Crafting Space	Decoration Shelves	Dishwasher	Filing Cabinets	Guest Book	Hall Gathering Place	Beverage Maker	Hymn Board	Hymn Book Shelf	Kitchen Island	Piano and Organ	Seat for Resting
Phoning																							
Notifying			X						X	X									X				
Playing				X	X																		
Praying																			X				
Waiting																							
Greeting										X						X							
Cooking																		X			X		
Serving																							
Drinking																	X						
Eating																	X						
Teaching					X												X						
Watching (Children)				X	X																		
Cleaning (Dishes)														X									
Making (Crafts)												X											
Making (Documents)																							
Performing (A Sermon)																							
Performing (Communion)								X															
Performing (Baptism)		X																					
Performing (Music)																						X	
Practicing (Music)						X																X	
Storing (Machinery)																							
Storing (Food)																							
Storing (Kitchen Tools)																					X		
Storing (Chairs and Tables)																							
Storing (Ritual Items)		X						X															
Storing (Sheet Music)						X														X		X	
Storing (Decorations)	X												X										
Storing (Instruments)						X																X	
Storing (Clothing)							X																
Storing (Custodial Items)																							
Storing (Cleaning Supplies)																					X		
Storing (Community Items)											X	X											
Storing (Craft Supplies)			X								X												
Storing (Books)			X																	X			
Storing (Files)															X								
Storing(Office Supplies)																							
Working (Computer)																							
Working (Paperwork)																							
Meeting (Large Group)																	X						
Meeting (Small Group)																							
Meeting (Few People)																							
Sitting (Worship)																							
Sitting (Short Term)																							X
Sitting (Long Term)																	X						

	Allows Light	Allows Sight	Allows Travel	Is Togglable	Has Covering	R/I Before	R/I After	Height	Width	Depth	Floor Distance	Center Distance	Orientation	Change in Elevation
Narthex Window	TRUE	FALSE	FALSE	FALSE	TRUE	0.0135	N/A	1016mm	406mm	152mm	838mm	2941mm	-90	N/A
Front Door	TRUE	TRUE	TRUE	FALSE	TRUE	0.0050	N/A	1981mm	1600mm	152mm	0mm	1368mm	180	0
Loft Window	TRUE	FALSE	FALSE	FALSE	TRUE	0.2263	N/A	2870mm	2159mm	152mm	431mm	1206mm	180	N/A
Sanctuary Window	TRUE	TRUE	FALSE	TRUE	TRUE	0.0377	0.1878	609mm	1828mm	152mm	1803mm	6124mm	90	N/A
Hall Window	TRUE	FALSE	FALSE	FALSE	TRUE	0.0136	N/A	1193mm	609mm	152mm	1244mm	5063mm	180	N/A
Hall Fire Exit	FALSE	FALSE	TRUE	FALSE	TRUE	N/A	N/A	2006mm	812mm	152mm	0mm	7616mm	0	0
Kitchen Side Entrance	TRUE	TRUE	TRUE	FALSE	FALSE	N/A	N/A	2641mm	1981mm	101mm	0mm	6066mm	N/A	0
Passthrough Window	TRUE	TRUE	FALSE	FALSE	FALSE	N/A	N/A	1092mm	1981mm	101mm	1041mm	4074mm	N/A	N/A
Storage Window	TRUE	TRUE	FALSE	FALSE	TRUE	0.0133	N/A	558mm	2438mm	152mm	1625mm	2844mm	-90	N/A
Office Closet Door	FALSE	FALSE	TRUE	FALSE	TRUE	N/A	N/A	2286mm	457mm	101mm	0mm	2254mm	N/A	0
Basement Fire Exit	FALSE	FALSE	TRUE	FALSE	TRUE	N/A	N/A	2006mm	812mm	152mm	0mm	4659mm	0	2052mm
Narthex Stairs	TRUE	TRUE	TRUE	FALSE	FALSE	N/A	N/A	2736mm	685mm	101mm	0mm	1953mm	N/A	2736mm
Basement Stairs	TRUE	TRUE	TRUE	FALSE	FALSE	N/A	N/A	1824mm	1219mm	101mm	0mm	558mm	N/A	1824mm
Office Window	TRUE	TRUE	FALSE	TRUE	TRUE	0.0213	0.0108	558mm	2438mm	152mm	1625mm	2606mm	-90	N/A

Table 5.2: Interface Data (By Author).

	Data Point Amount	Relative Illuminance	Background Noise	Average Temperature	Floor Material A	Wall Material A	Ceiling Material A	Floor Material B	Wall Material B	Ceiling Material B
Entry	3	0.3225	26.3db	17°C	Vinyl Tile	Drywall	Drywall	Carpet	Drywall	Ceiling Tile
Narthex	3	0.0960	26db	17.5°C	Carpet	Drywall	Drywall	Vinyl Tile	Drywall	Ceiling Tile
Hallway	4	0.0818	24.5db	11.5°C	Carpet	Drywall	Drywall	Carpet	Paneling	Ceiling Tile
Office	4	0.1408	26db	17°C	Carpet	Drywall	Drywall	Vinyl Tile	Paneling	Ceiling Tile
Sanctuary	3	0.0525	26.6db	17°C	Vinyl Tile	Paneling	Ceiling Tile	Carpet	Drywall	Wood
Hall	3	0.1730	24.5db	17°C	Vinyl Tile	Paneling	Ceiling Tile	Concrete	Drywall	Paneling
Washroom	6	0.0454	26db	17.3°C	Vinyl Tile	Drywall	Drywall	Vinyl Tile	Tile	Ceiling Tile
Kitchen	4	0.0946	25.5db	17.7°C	Vinyl Tile	Paneling	Ceiling Tile	Vinyl Tile	Drywall	Drywall
Boiler Room	3	0.0	28db	18°C	Concrete	Concrete	Concrete	Vinyl Tile	Concrete	None
Storage	7	0.0634	26.4db	15.5°C	Carpet	Drywall	Drywall	Concrete	Paneling	Ceiling Tile
Garden Storage	2	0.0	22db	14.5°C	Concrete	Concrete	Concrete	Concrete	None	None
Landing	5	0.2227	24.8db	17°C	Vinyl Tile	Drywall	Drywall	Vinyl Tile	Paneling	Ceiling Tile
Sunday School Room	2	0.1083	25.5db	14.5°C	Carpet	Drywall	Drywall	Concrete	Paneling	Ceiling Tile
Loft	1	0.1089	24db	17°C	Carpet	Drywall	Ceiling Tile	Carpet	Drywall	Ceiling Tile
Supply Closet	2	0.0	24db	14.5°C	Concrete	Drywall	Drywall	Concrete	None	Ceiling Tile
Office Closet	3	0.0	24db	17°C	Carpet	Drywall	Ceiling Tile	Carpet	Drywall	Ceiling Tile
Maintanance Corridor	1	0.0	29db	11°C	Concrete	Concrete	None	Concrete	None	None
Basement Hallway	2	0.0344	24.5db	17°C	Carpet	Drywall	Drywall	Vinyl Tile	Paneling	Ceiling Tile
Secretary's Office	1	0.1671	27db	16°C	Carpet	Drywall	Drywall	Carpet	Drywall	Drywall
Minister's Office	3	0.1578	25.6db	17°C	Carpet	Drywall	Drywall	Vinyl Tile	Drywall	Drywall
Cloak Room	2	0.1335	27db	18°C	Carpet	Drywall	Drywall	Vinyl Tile	Drywall	Drywall
Parlour	1	0.3100	29db	17°C	Carpet	Drywall	Drywall	Carpet	Drywall	Drywall
Open Storage	1	0.0	27db	17°C	Concrete	Drywall	Ceiling Tile	Vinyl Tile	Paneling	Ceiling Tile

Table 5.3: Aggregate Environmental Data (By Author).

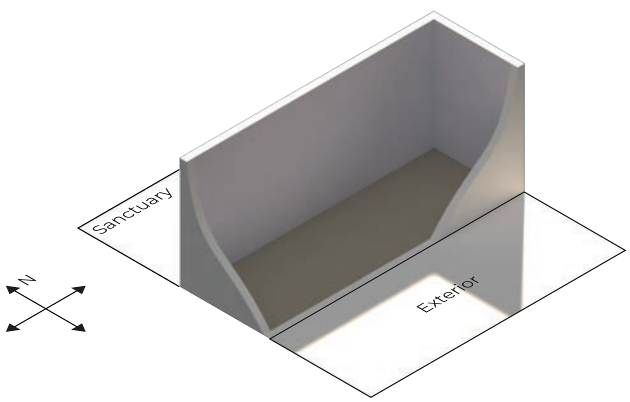
	Entry	Narthex	Hallway	Office	Sanctuary	Hall	Washroom	Kitchen	Boiler Room	Storage	Garden Storage	Landing	Sunday School Room	Loft	Supply Closet	Office Closet	Maintanance Corridor	Basement Hallway	Secretary's Office	Minister's Office	Cloak Room	Parlour	Open Storage
Phoning				■				■											■	■			
Notifying	■	■				■							■					■					
Playing													■										
Praying					■																		
Waiting		■																					
Greeting	■	■																					
Cooking								■															
Serving						■		■														■	
Drinking						■																■	
Eating						■																■	
Teaching					■								■									■	
Watching (Children)													■										
Cleaning (Dishes)								■															
Making (Crafts)													■									■	
Making (Documents)				■															■	■			
Performing (A Sermon)					■																		
Performing (Communion)					■																		
Performing (Baptism)					■																		
Performing (Music)					■																		
Practicing (Music)					■																	■	
Storing (Machinery)									■								■						
Storing (Food)								■															
Storing (Kitchen Tools)								■		■													
Storing (Chairs and Tables)				■		■				■				■									■
Storing (Ritual Items)		■			■			■															
Storing (Sheet Music)					■																		
Storing (Decorations)										■													
Storing (Instruments)					■																		
Storing (Clothing)		■														■				■	■	■	
Storing (Custodial Items)				■							■								■	■			
Storing (Cleaning Supplies)							■	■															
Storing (Community Items)						■				■			■									■	
Storing (Craft Supplies)										■			■										
Storing (Books)				■									■						■	■			■
Storing (Files)				■											■	■			■	■			
Storing(Office Supplies)				■									■		■	■			■	■			
Working (Computer)				■															■	■			
Working (Paperwork)				■									■						■	■			
Meeting (Large Group)					■	■																	
Meeting (Small Group)				■	■	■							■						■	■		■	
Meeting (Few People)	■	■		■	■	■							■						■	■		■	
Sitting (Worship)					■									■									
Sitting (Short Term)				■	■	■						■	■	■					■	■	■	■	
Sitting (Long Term)				■	■	■							■	■					■	■		■	

Table 5.4: Aggregate Active Affordance Data (By Author).

In theory, with enough context data it would be possible to automate this process of affordance matching with a high degree of consistency with human design decisions.

As an example of how this pattern matching process works we can add the objects back into a narthex space on the next few pages. (Figure 5.2) Starting with the requirements that the space needs in order to perform it’s function properly, we add the materials to the space defined by the generative algorithm. Next we add the openings, making sure to include the correct kind of access between all the required adjacencies. Finally we add objects to the space, choosing objects which have both previously been found in this type of space in the dataset, and also fill the requirements for active affordances.

This idea of pairing the generated data about space and affordance requirements from the schematic design with the pattern language of objects and interfaces collected from the case studies allows for the data driven population of a generated floor plan with objects. In the next chapter, this process will be applied to a whole building by hand to generate a finished design.



Narthex

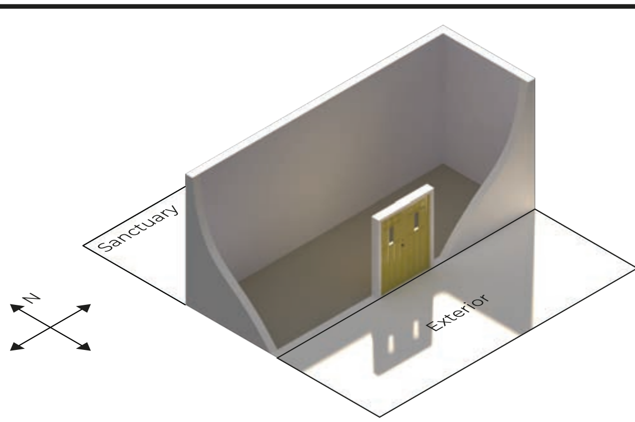
Active Affordances
- -Meeting Few People
- -Storing Clothing
- -Greeting
- -Waiting
- -Notifying

Access
- -Sanctuary
- -Exterior
- -Windows, 1, Eastern Site

Environmental Affordances
- -Relative Illuminance of 0.0960
- -Carpeted Floor
- -Drywall Walls
- -Ceiling Tile Ceiling

Narthex Steel Double Door

Active Affordances	Environmental Affordances	Context
-None	-Relative Illuminance of 0.0054 -Allows Light -Allows Sight -Allows Access -Connects Exterior and Narthex	-Height, 1981mm -Width, 1600mm -Depth, 152mm -Center Distance, 1368mm -Frame, Wood -Infill, Steel/Glass



Active Affordances
- -Meeting Few People
- -Storing Clothing
- -Greeting
- -Waiting
- -Notifying

Access
- -Sanctuary
- -Exterior
- -Windows, 1, Eastern Site

Environmental Affordances
- -Relative Illuminance of 0.0960
- -Carpeted Floor
- -Drywall Walls
- -Ceiling Tile Ceiling

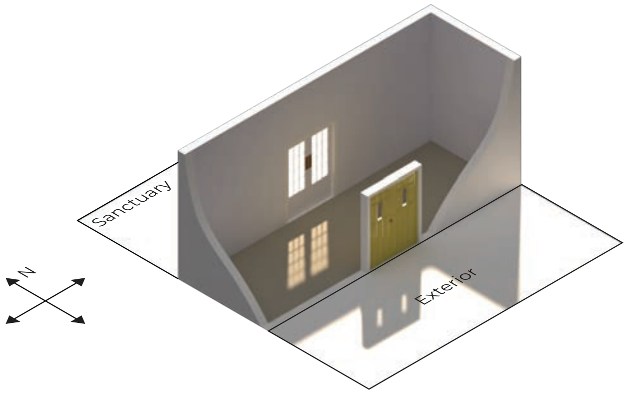
Figure 5.2: Affordance Mapping Example (By Author).

Sanctuary Wood Double Door

Active Affordances
- -None

Environmental Affordances
- -Relative Illuminance of 0.0096
- -Allows Light
- -Allows Sight
- -Allows Access
- -Connects Sanctuary and Narthex

Context
- -Height, 1955mm
- -Width, 1524mm
- -Depth, 152mm
- -Center Distance, 1780mm
- -Frame, Wood
- -Infill, Wood /Glass



Active Affordances
- -Meeting Few People
- -Storing Clothing
- -Greeting
- -Waiting
- -Notifying

Access
- -Sanctuary
- -Exterior
- -Windows, 1, Eastern Site

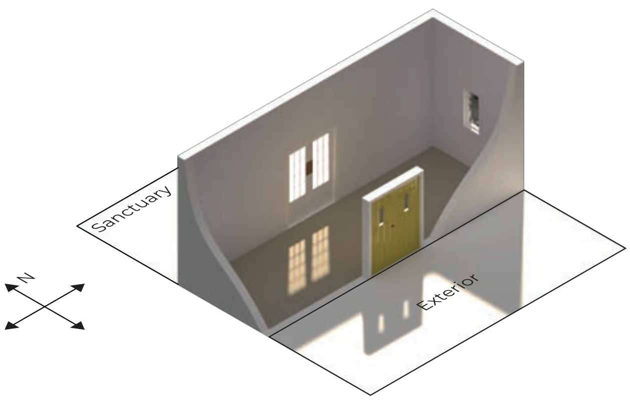
Environmental Affordances
- -Relative Illuminance of 0.0960
- -Carpeted Floor
- -Drywall Walls
- -Ceiling Tile Ceiling

Narthex Frosted Window

Active Affordances
- -None

Environmental Affordances
- -Relative Illuminance of 0.0135
- -Allows Light
- -Connects Exterior and Narthex

Context
- -Height, 1016mm
- -Width, 406mm
- -Depth, 152mm
- -Floor Distance, 838
- -Center Distance, 2941mm
- -Material, Plastic and Glass



Active Affordances
- -Meeting Few People
- -Storing Clothing
- -Greeting
- -Waiting
- -Notifying

Access
- -Sanctuary
- -Exterior
- -Windows, 1, Eastern Site

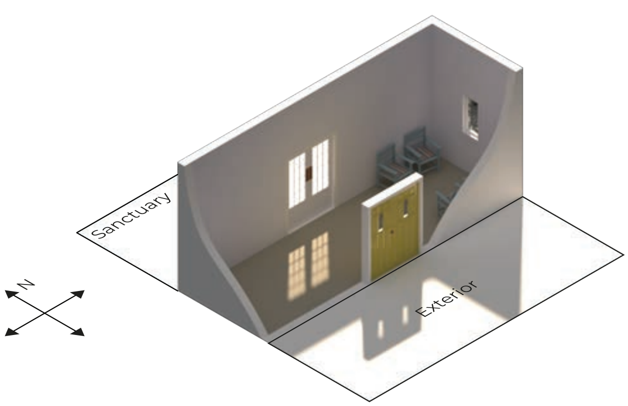
Environmental Affordances
- -Relative Illuminance of 0.0960
- -Carpeted Floor
- -Drywall Walls
- -Ceiling Tile Ceiling

Narthex Meeting and Waiting Area

Active Affordances
- -Meeting Few People
- -Waiting

Environmental Affordances
- -None

Context
- -Found in Narthex and Entry



Active Affordances
- -Meeting Few People
- -Storing Clothing
- -Greeting
- -Waiting
- -Notifying

Access
- -Sanctuary
- -Exterior
- -Windows, 1, Eastern Site

Environmental Affordances
- -Relative Illuminance of 0.0960
- -Carpeted Floor
- -Drywall Walls
- -Ceiling Tile Ceiling

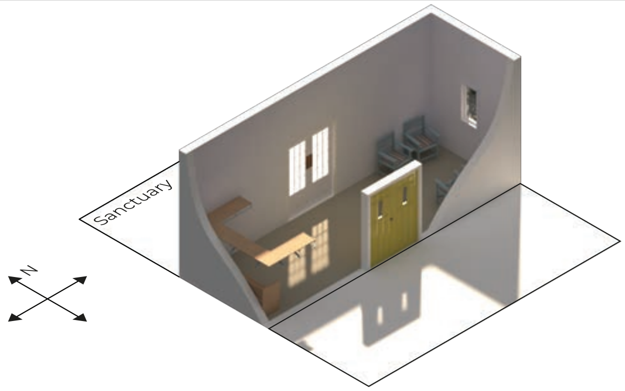
Figure 5.2: Affordance Mapping Example (By Author).

Undressing Area

Active Affordances
- -Storing Clothing
- -Waiting

Environmental Affordances
- -None

Context
- -Found in Narthex



Active Affordances
- -Meeting Few People
- -Storing Clothing
- -Greeting
- -Waiting
- -Notifying

Access
- -Sanctuary
- -Exterior
- -Windows, 1, Eastern Site

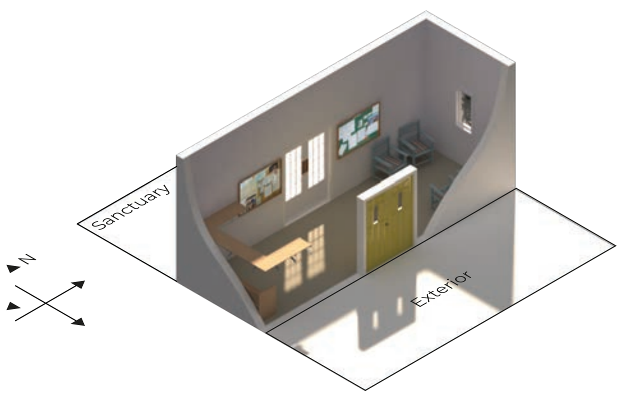
Environmental Affordances
- -Relative Illuminance of 0.0960
- -Carpeted Floor
- -Drywall Walls
- -Ceiling Tile Ceiling

Community Cork Board

Active Affordances
- -Notifying

Environmental Affordances
- -None

Context
- -Found in Narthex and Hall



Active Affordances
- -Meeting Few People
- -Storing Clothing
- -Greeting
- -Waiting
- -Notifying

Access
- -Sanctuary
- -Exterior
- -Windows, 1, Eastern Site

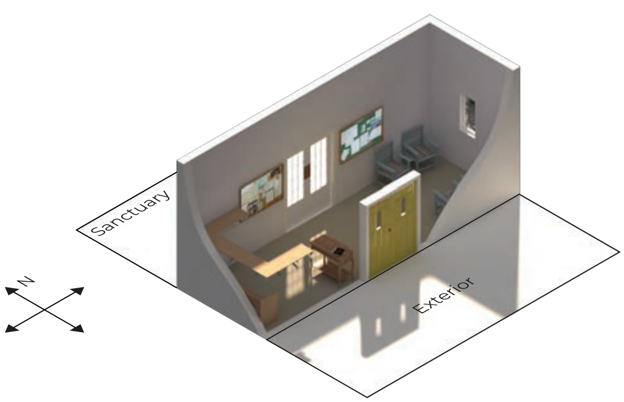
Environmental Affordances
- -Relative Illuminance of 0.0960
- -Carpeted Floor
- -Drywall Walls
- -Ceiling Tile Ceiling

Welcome Table

Active Affordances
- -Greeting

Environmental Affordances
- -None

Context
- -Found in Narthex and Entry
- -Holds Bulletins



Active Affordances
- -Meeting Few People
- -Storing Clothing
- -Greeting
- -Waiting
- -Notifying

Access
- -Sanctuary
- -Exterior
- -Windows, 1, Eastern Site

Environmental Affordances
- -Relative Illuminance of 0.0960
- -Carpeted Floor
- -Drywall Walls
- -Ceiling Tile Ceiling

Figure 5.2: Affordance Mapping Example (By Author).

Design Proposal And Process

The final plan was generated using the data collected from the previous case studies as well as some user generated data guided by my own intuition. This final generation was done with 13 spaces as opposed to the 10 that were used in the smaller scale tests.

For this example, the solar access cost algorithm had stricter requirements for surface area than in the previous tests, using average surface area of the whole wall as the goal rather than just glazed surface area. The majority of spaces met this requirement except the sanctuary. It missed the target by about 25% due to restrictions on area and height forcing the algorithm to compromise on the space by optimizing the correct proportions at the cost of less surface area.

The final plan managed to replicate the two wing church design similar to Larchwood Memorial. The design places the narthex in the center of the two wings allowing for entry into the church space and the hall space from that single point. Bringing the Sanctuary space forward allows for an administrative wing that shares a central storage room with the sanctuary and the hall.

Openings and furnishings were placed in this plan using the data itself as a driving factor. The types of openings used, and the location of those openings were matched as closely as possible to the spaces found in the final design, and the locations that those openings would be found relative to the center of the space. Materiality was also matched to be a type of material found in that particular space in the dataset.

The location of columns and beams within the building, and the final shape of the roof still relies entirely on human intuition. The precise application of this data is also guided by human intuition while it is not yet managed with an automated generative process. Further research into the type of information needed and the feasibility of the full automation of the affordance matching process remains to be explored.

Area Cost

	Upper	Lower
Narthex	31.62 mSq	24.50 mSq
Hall	118.86 mSq	77.19 mSq
Sanctuary	139.13 mSq	89.23 mSq
Kitchen	29.99 mSq	18.30 mSq
Parlour	70.61 mSq	67.43 mSq
Office	12.88 mSq	9.07 mSq
Minister’s Office	18.9 mSq	12.91 mSq
Washroom 1	5.13 mSq	3.86 mSq
Washroom 2	5.13 mSq	3.86 mSq
Storage	21.38 mSq	12.25 mSq
Open Storage	23.07 mSq	19.63 mSq
Boiler Room	11.54 mSq	7.75 mSq
Back Hallway	31.87 mSq	17.00 mSq

	Plan Cost
Narthex	0
Hall	0
Sanctuary	0
Kitchen	0
Parlour	8.65
Office	0.45
Minister’s Office	0
Washroom 1	0
Washroom 2	0
Storage	0
Open Storage	2.41
Boiler Room	0
Back Hallway	0

Final Plan

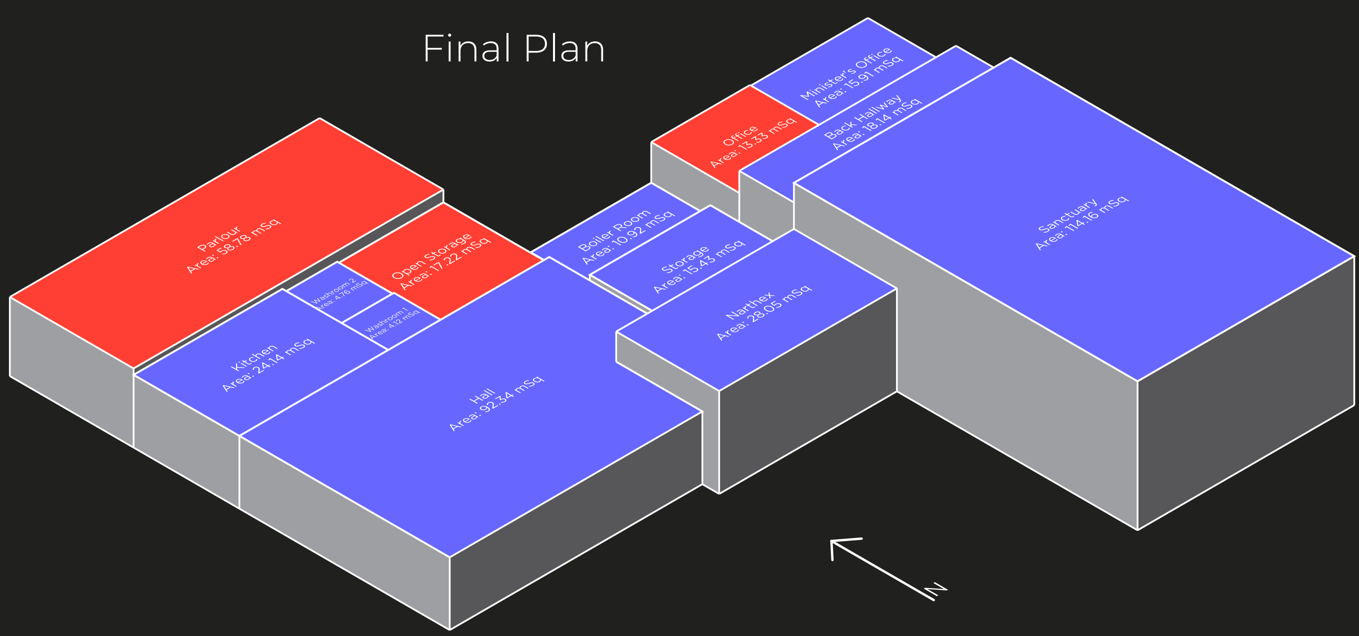
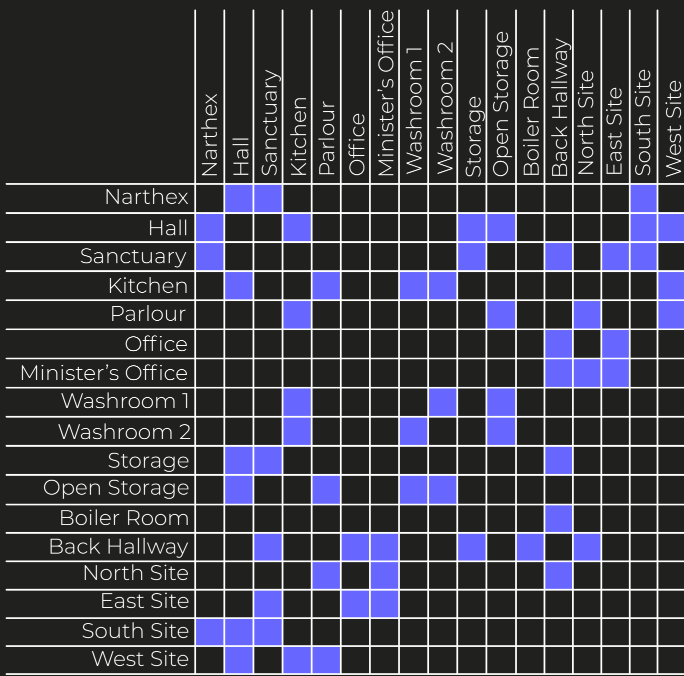


Figure 6.1: Final Schematic Design. Area Optimization Data (By Author).

Adjacency Cost



	Plan Cost
Narthex	3 - 3 = 0
Hall	5 - 5 = 0
Sanctuary	5 - 4 = 1
Kitchen	5 - 5 = 0
Parlour	3 - 3 = 0
Office	2 - 1 = 1
Minister's Office	3 - 3 = 0
Washroom 1	3 - 3 = 0
Washroom 2	3 - 3 = 0
Storage	3 - 2 = 1
Open Storage	4 - 4 = 0
Boiler Room	1 - 0 - 1
Back Hallway	6 - 4 = 2

Final Plan

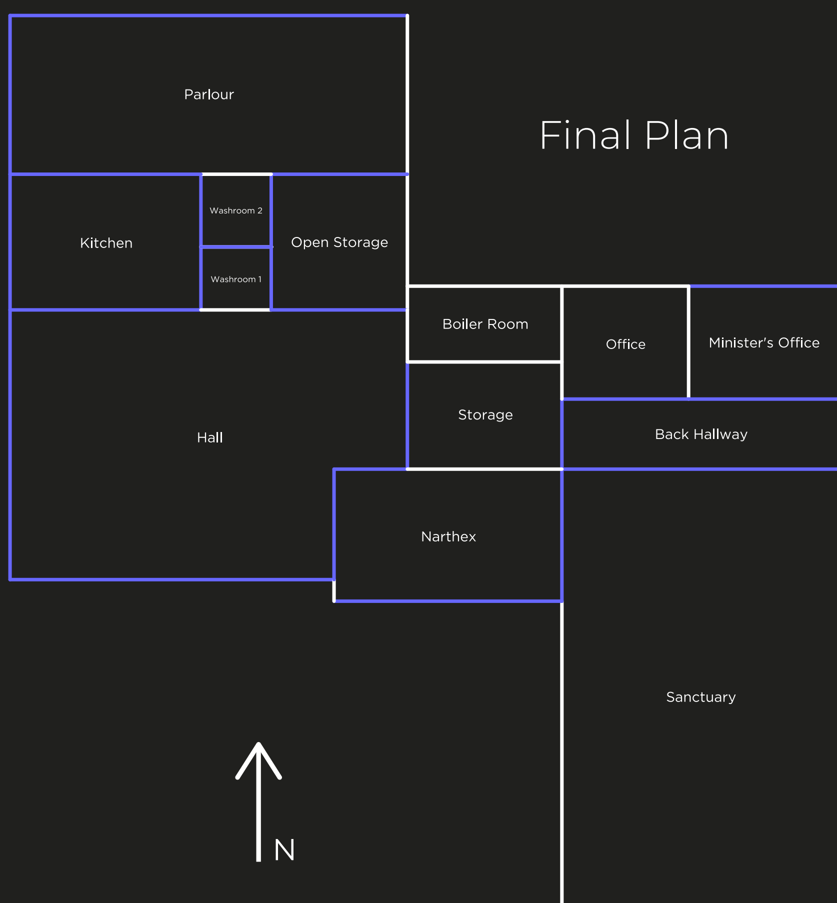


Figure 6.2: Final Schematic Design. Adjacency Optimization Data (By Author).

Ratio Cost

	Upper	Lower
Narthex	0.6833	0.4707
Hall	0.7254	0.6063
Sanctuary	0.7092	0.5520
Kitchen	0.8027	0.6188
Parlour	0.7052	0.6146
Office	0.8172	0.6564
Minister’s Office	0.7661	0.7276
Washroom 1	0.7191	0.6806
Washroom 2	0.7191	0.6806
Storage	0.6610	0.6124
Open Storage	0.8314	0.7084
Boiler Room	0.6334	0.4779
Back Hallway	0.6052	0.4124

	Plan Cost
Narthex	0
Hall	0
Sanctuary	0
Kitchen	0
Parlour	0.2152
Office	0.0742
Minister’s Office	0
Washroom 1	0.1767
Washroom 2	0.2476
Storage	0.0324
Open Storage	0.1651
Boiler Room	0
Back Hallway	0.1604

Final Plan

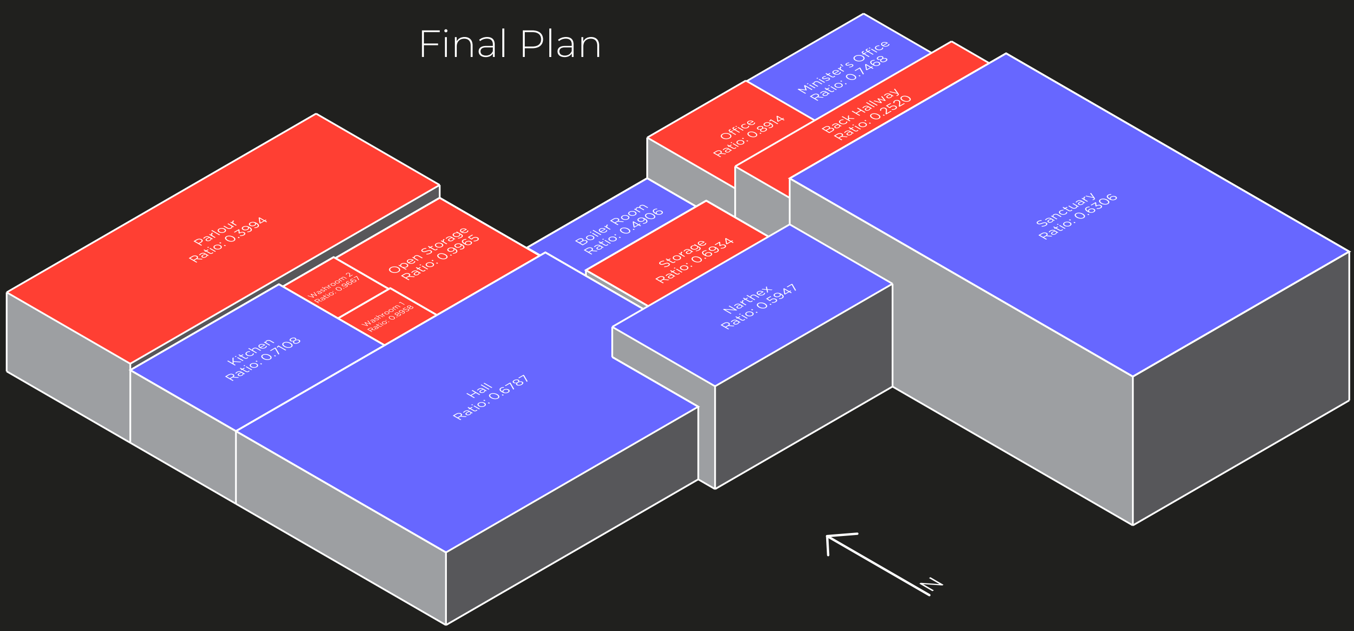


Figure 6.3: Final Schematic Design. Ratio of Length to Width Optimization Data (By Author).

Shape Cost

	Plan Cost
Narthex	0
Hall	5
Sanctuary	0
Kitchen	0
Parlour	0
Office	0
Minister's Office	0
Washroom 1	0
Washroom 2	0
Storage	0
Open Storage	0
Boiler Room	0
Back Hallway	0

Final Plan

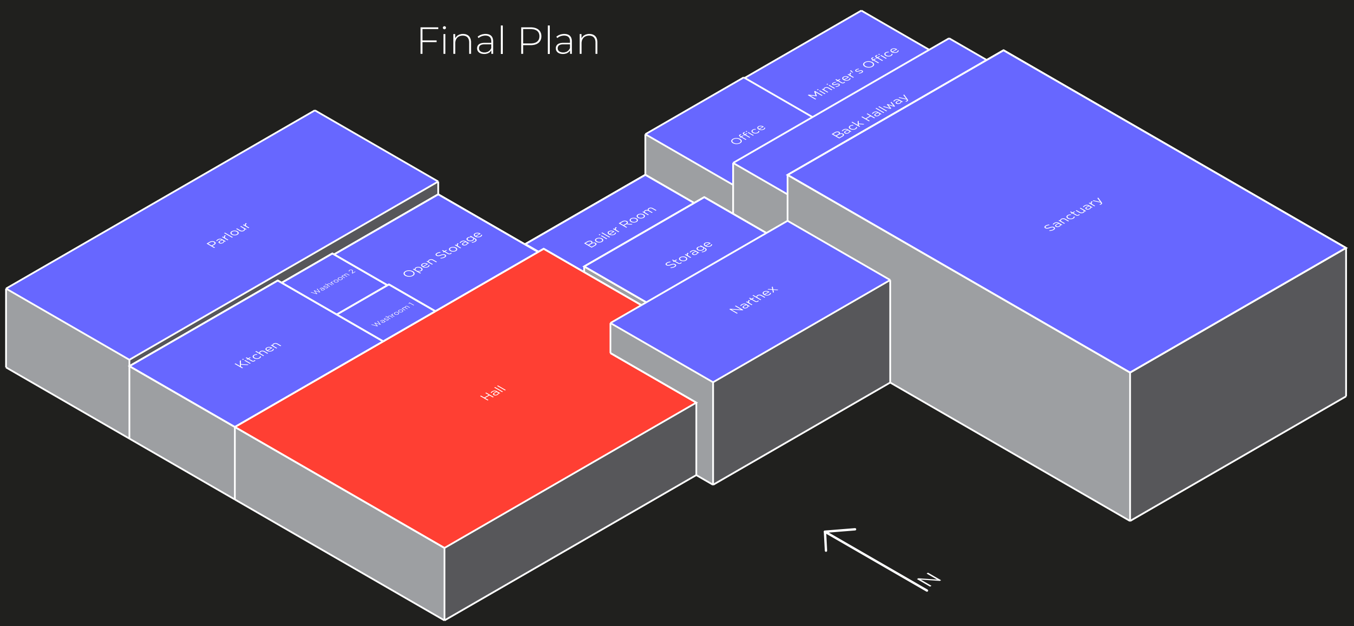


Figure 6.4: Final Schematic Design. Shape and Convexity Optimization Data (By Author).

Height Cost

	Upper	Lower
Narthex	2.59m	2.02m
Hall	2.56m	2.39m
Sanctuary	6.46m	4.11m
Kitchen	2.58m	2.36m
Parlour	2.62m	1.42m
Office	2.38m	1.29m
Minister’s Office	2.58m	2.47m
Washroom 1	2.40m	2.33m
Washroom 2	2.40m	2.33m
Storage	2.84m	2.53m
Open Storage	2.60m	2.39m
Boiler Room	2.31m	2.05m
Back Hallway	2.86m	2.56m

	Plan Cost
Narthex	0.90
Hall	0
Sanctuary	0
Kitchen	0
Parlour	0.06
Office	1.26
Minister’s Office	1.03
Washroom 1	0
Washroom 2	0
Storage	0
Open Storage	0.002
Boiler Room	0
Back Hallway	1.53

Final Plan

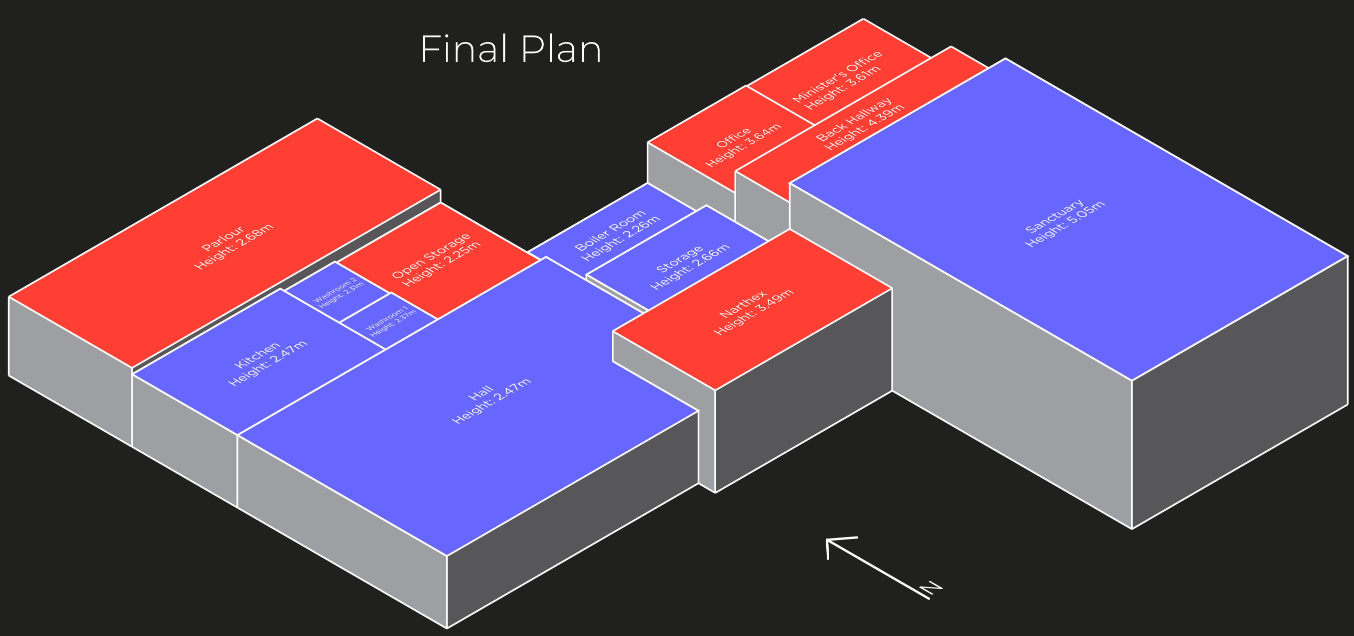


Figure 6.5: Final Schematic Design. Height Optimization Data (By Author).

Solar Access Cost

	Upper	Lower
Narthex	---	35.30mSq
Hall	---	42.68mSq
Sanctuary	---	221.25mSq
Kitchen	---	10.77mSq
Parlour	---	11.90mSq
Office	---	15.9mSq
Minister's Office	---	17.2mSq
Washroom 1	---	0.00mSq
Washroom 2	---	0.00mSq
Storage	---	8.84mSq
Open Storage	---	0.00mSq
Boiler Room	---	0.00mSq
Back Hallway	---	11.76mSq

	Plan Cost
Narthex	0
Hall	0
Sanctuary	62.84
Kitchen	0.56
Parlour	0
Office	15.9
Minister's Office	0
Washroom 1	0
Washroom 2	0
Storage	8.42
Open Storage	0
Boiler Room	0
Back Hallway	2.38

Final Plan

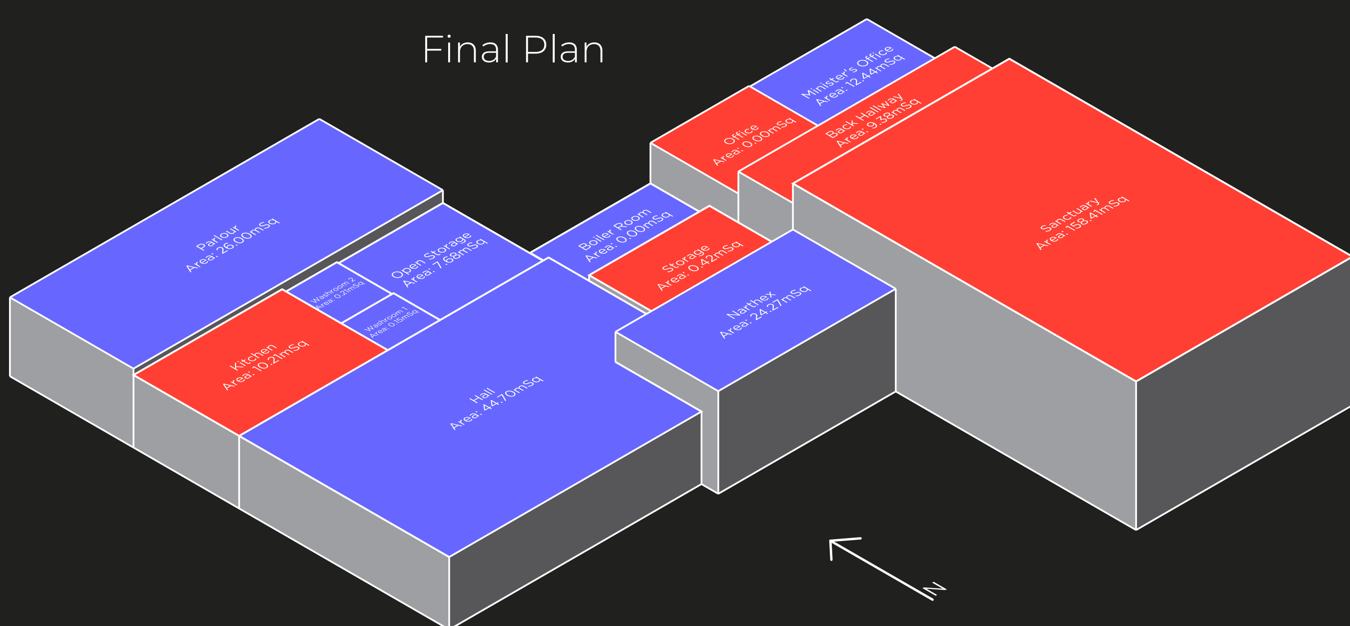


Figure 6.6: Final Schematic Design. Solar Access Optimization Data (By Author).


Figure 6.7: Final Site Plan and Programmatic Layout (By Author).

Figure 6.8: Final Floor Plan (By Author).

Figure 6.9: Final Sanctuary Vignette (By Author).

This final vignette shows what the sanctuary of this generated church could look like after it has been populated with objects and materials found within the case study buildings (Figure 6.9). The materials, furnishings, lighting, and openings into the space were taken directly from objects found in the case studies. They were chosen as examples of objects which satisfied certain active affordances required for that type of space.

The objects were applied to the space generally, using data about their relationship with the center of the room to inform their placement. The application of these objects to satisfy the requirements of space does currently need the subjective intuition of a human being to ensure that the space actually functions as intended. However, the addition of further data regarding the location and use of objects within space could allow for a more automated system for the application of objects and openings in the future.

Future Experiments

There are two main areas which I believe should be explored in the future. The first is some further refinements to the generative system and the development of the affordance matching algorithm.

There are some limitations to the current implementation of the generative algorithm. One is that it was created in Grasshopper, and that makes the simulated annealing process incredibly slow. Another is that it is missing some useful functionality, namely the ability to define a building footprint and the ability to generate building massing which spans more than a single floor. To amend this, a new version of the system should be created as a Grasshopper plugin that can run in a faster environment.

Additionally, more research should be done into creating a BIM-like affordance matching algorithm which is capable of using a database of objects to populate schematic designs. The research and creation of parametric archetypal objects which could be altered to fit the affordances of space needed for a single user would be a powerful extension of the current use of affordance theory to categorize space within a building.

The second area that I believe should be explored is the inclusion of psychometric data. Currently the validity of the decisions made by the generative system are based upon the intuitions of the operator. A different and interesting way to approach this is to allow for value judgments based on psychological effects that choices made by that AI agent will have on the users of that space.

Collecting information about how spaces make users feel would allow for a comparison between different objects which afford the same things. The choice of which solution to use could be made by attempting to optimize for a particular emotion, or state of consciousness in a hypothetical user. The initial exploration into this area is documented in Appendix 1.

Final Remarks

This thesis explored the use of Gibson's theory of affordances to categorize and describe the essential elements of architecture. It also explored how in conjunction with space syntax, this theory can be used to extract data which is usable by a computer algorithm to quantify architectural space. Specific elements and implementations of architectural affordance were shown to be able to be gathered into a type of pattern language which on it's own can be used to apply active affordances onto schematic designs.

Taking elements from previous generative design work, this thesis then detailed an iterative generative algorithm taking six cost variables into account: area, adjacency, ratio, shape, height, and solar access. This algorithm was built upon Rhino and Grasshopper as a framework and used an optimization solution based on iterative simulated annealing in order to converge on an ideal solution.

As a method of data collection a sensor array was developed for this thesis, capable of collecting information about light, sound, and temperature in space. This data was organized using a system of nodes and edges showing the physical properties of architectural spaces within a building and the properties of the interfaces including windows and doors that allow information to pass between spaces.

This thesis used case studies of united churches in order to explore the application of this theoretical system. Affordances were used as a model of this building typology to allow physical aspects of space to be extracted from the case studies, and then organized as a series of nodes and edges which could then be fed into the generative system. Objects and interfaces within these case studies were also cataloged and kept as a pattern language of space which could then be applied to the schematic designs later to reintroduce affordances of space back into the massing and generate a completed architectural design.

Dominion Chalmers in Ottawa was used as a case study to explore the typology of the united church on the whole, and to generate a list of active affordances which could be found in these buildings. St. Steven's, Larchwood Memorial, and St. Mark's were used as similar scale examples of united church spaces whose data could be used to test the generative system.

Examples of this system replicating the building typology found in these case studies were explored showing how this generative system is able to generate schematic designs similar in form and function to the ones examined in the case studies. A larger scale final design was then generated as an example to be used with the pattern matching approach of populating space with affordance objects.

Objects found and documented from the case studies were matched with the schematic designs generated for the final design in order to create a set of final plans with openings between spaces and the outdoors, as well as furnishings. As of now this process still requires a fair amount of input and intuition from the architect. In future this process could be automated given more object data and context.

Limitations to this approach as it has been explored in this thesis are the speed of the calculations as they are implemented in Grasshopper, the implementation and automation of the pattern matching process, and the inclusion of psychometric data in order to allow for value judgments between different possible design decisions to be taken into account. Preliminary research into how this psychometric data could be captured and analyzed in order to match emotions with physical properties of space was undertaken as a part of the research for this thesis.

Overall this thesis project presents an application of affordances analysis and generative design. It highlights the potential for allowing generative systems the grammar to express and manipulate more functional and phenomenological aspects of architectural space. The application of affordances to generative systems within architectural design is an area full of promise and therefore should be explored in future research.

Appendix One

Neural Networks and Psychometrics

In order for a system which can make value judgments based upon human emotions to work, it is necessary to be able to predict how an environment will effect the psychological state of it's occupants. As a part of this thesis, I already did some preliminary work into this subject through the use of neural networks.

The environmental variables that were chosen included natural light, artificial light, temperature, humidity, and background noise. The intention was to document five other variables related to the physical form of the space as well, such as room and glazing area, height, perimeter, and the longest chord. The data was to be captured using the device described in the beginning of this chapter as a means of testing that approach as well.

The psychometric variables used were from the EMOTIV Insight EEG headset. It could measure your brainwaves and determine various levels of 6 different emotions: Stress, relaxation, excitement, Interest, Focus, and Engagement. (Figure 9.1) Due to time and physical constraints, the use of these tools was not possible in the field. As a test of the theory in general, a neural network that could compute these values with a fake dataset of 10,000 different data points was developed regardless.

The dataset was constructed by randomly generating values from a normal distribution for the different environmental variables, and then applying these various 10 dimensional function to them in order to simulate the kind of complex relationship that the environmental variables might have with the psychometric variables in real life. That dataset was then used to train a neural network until it reached an accuracy rate of at least 90%.

In the most basic form of a neural network, there are three types of layer, the input layer, the hidden layer, and the output layer. (Figure 9.2) Layers are made up of nodes, and for each of the nodes in the previous layer, each node has a given weight and an activation function that takes into account all the previous nodes which are connected to it. We can put values into the input layer and then watch them be computed through the layers in the middle until we get a value on the output. The network is trained by comparing the output of the network to the actual answer we are expecting. We compare the two numbers using what’s called a Loss function, then we apply a method called backpropagation to change the weights of each layer in the network working backwards until the network is pushed a little closer to giving us the right answer. If you repeat this process thousands of times, eventually you’ll have taught a model to approximate any function even if you don’t know what the function itself was from the start.

The actual network that was trained on this data had ten nodes at the beginning for the ten environmental variables, and then one node at the end to represent one of the six psychometric variables. In the middle, there were four layers with one-thousand-twenty-four nodes each. (Figure 9.3) The network was trained six different times to create a predictive measure for each of the psychometric variables. After this, we could use the network as method of data analysis.

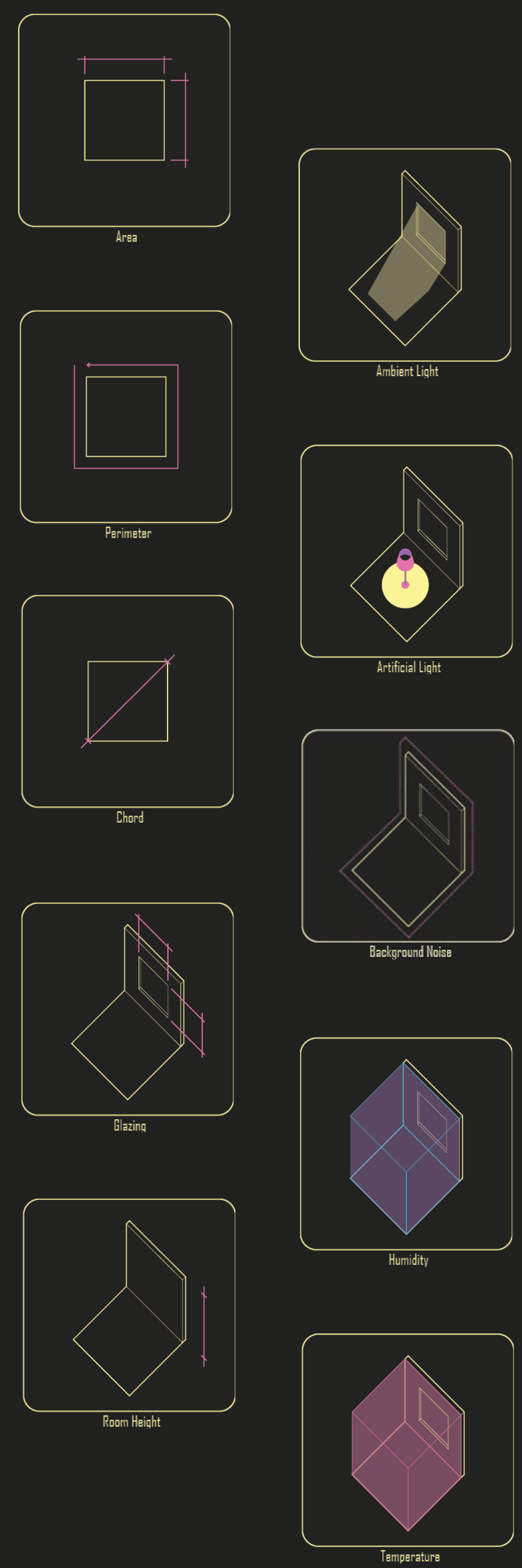


Figure 9.1 Physical variables used (Gagnon, 2021).

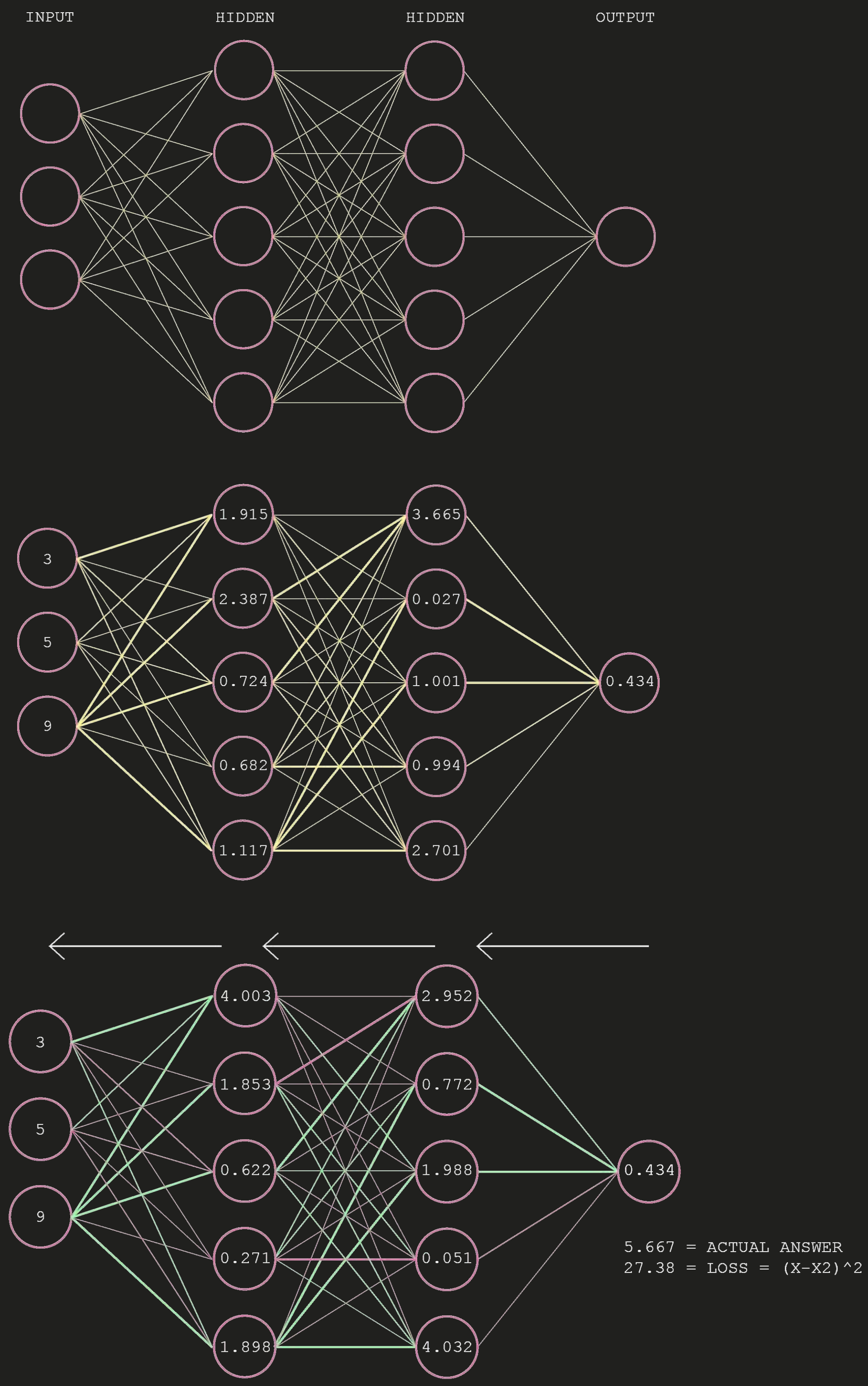


Figure 9.2 Backpropagation diagram (By Author).

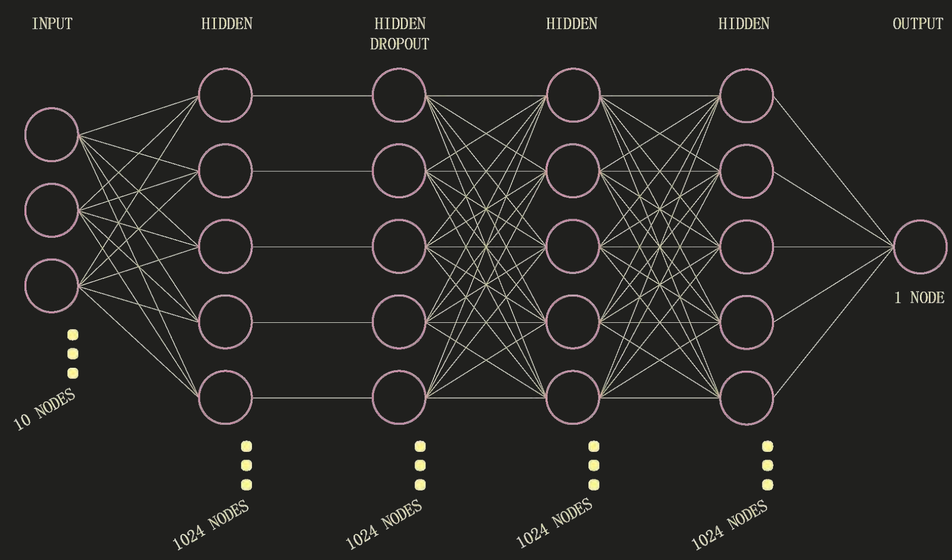
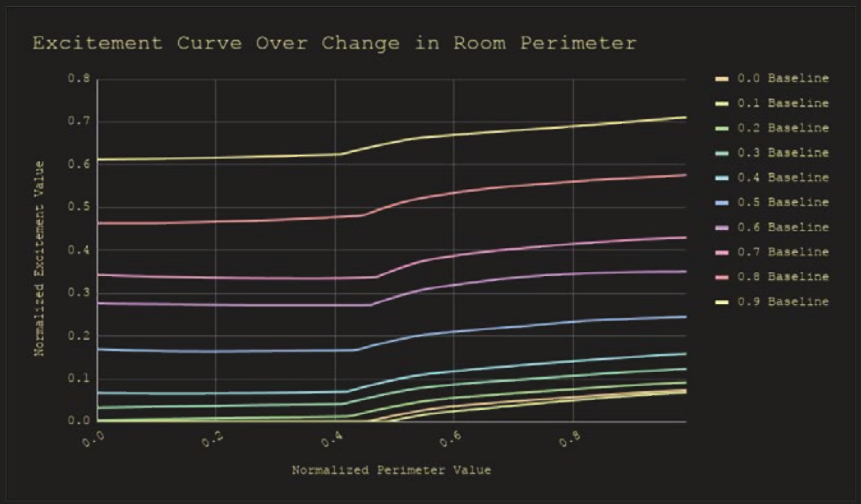


Figure 9.3 Final network diagram (By Author).



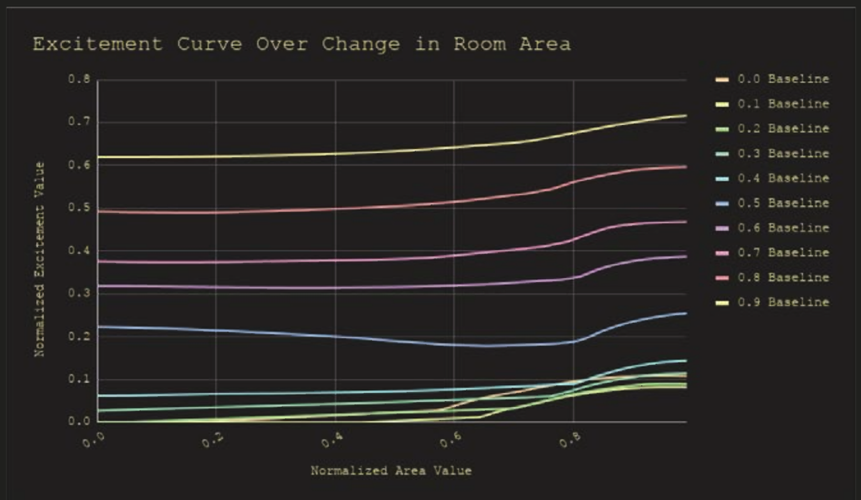


Figure 9.4 Example of output (By Author).

This network was able to learn the trends of the dataset across these ten different variables. A regression model used for analysis allows for fixing all the variables but one in a single position and then sweeping a single variable at a time over the space. As a result the landscape that is being regressed to using these models can be extracted and approximated. We can graph the outcomes and use those graphs to make informed predictions about how changing environmental variables affect the psychological state of the user. For example, in these graphs (Figure 9.4) you can see how excitement changes given a certain baseline, as you increase the level of background noise, the perimeter of the room, the height of the room, or the room area. This information lets us start to understand how changes to the variables in a space can directly affect the psychological state of the occupant.

Appendix Two

Grasshopper Definition Overview

This section of the book contains a detailed overview of the grasshopper definition used to generate the schematic design examples from earlier chapters. Explanations of the functionality are embedded in the definition itself through the grouping of related functions together, the naming of specific nodes to describe their use, and the annotation of various function groups to describe the overall function of that group.

Aside from nodes which come with an unmodified version of Grasshopper, there are a few key plugins which are essential to the functionality of this definition. Loop is the backbone of the simulation allowing for the iterative generation of geometry and continual processing of the floor plan required for the simulated annealing algorithm (Turiello, https://www.food4rhino.com/en/app/loop). Objectify allows for efficient data management in the definition making possible the creation of "structures" similar to those in C based programming languages. Groups of dissimilar variables can be easily routed through the program and contained within a single object. This is used for the representation of the floor plan, binding together geometry, text labels, and height values (Mahankali, https://www.food4rhino.com/en/app/objectify). Ladybug tools are also used for calculations related to solar exposure and vectors. (https://www.food4rhino.com/en/app/ladybug-tools).

For more information regarding the plugins' function, please refer to the relevant documentation provided by their authors.

Main Program Loop

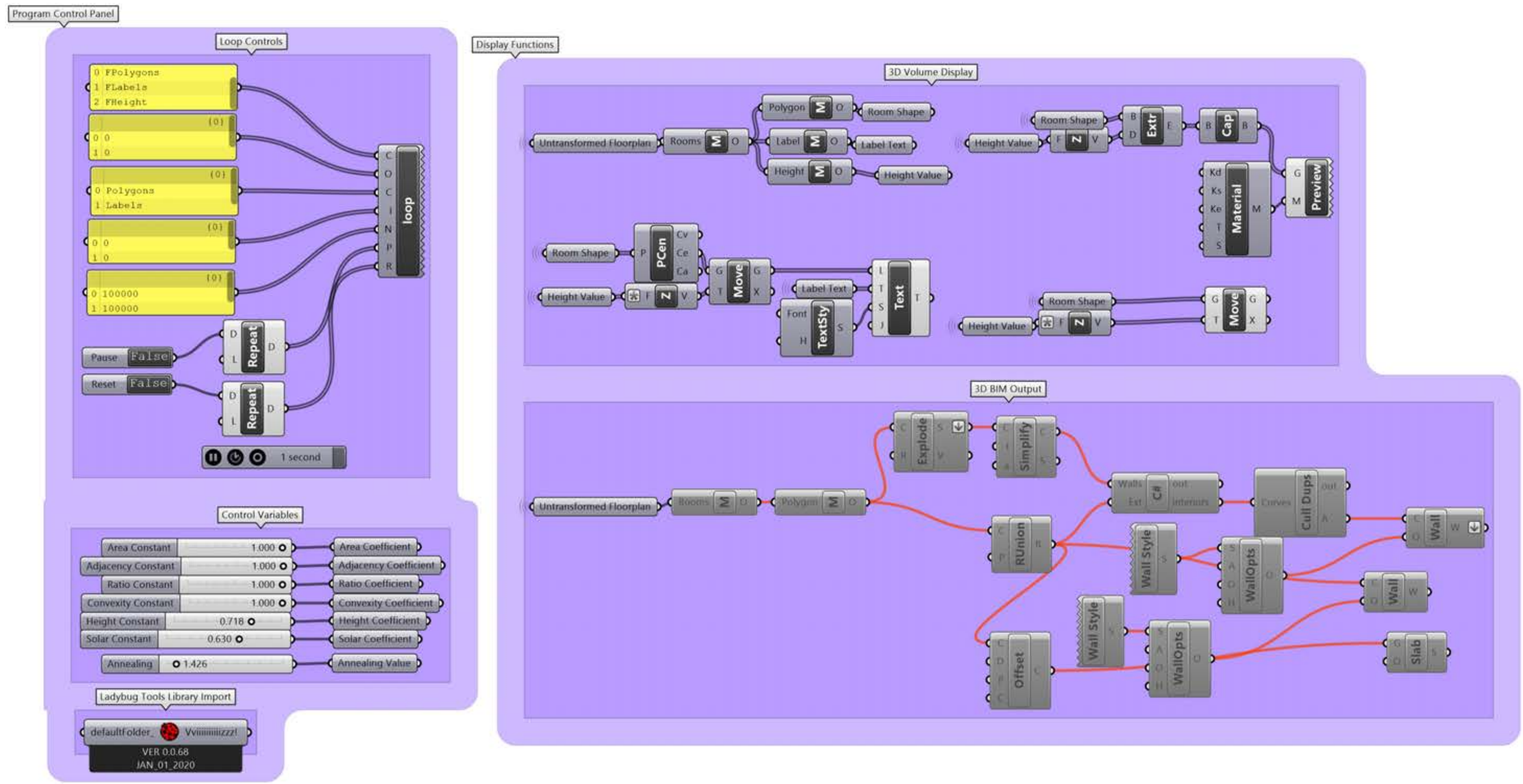


Figure 10.1 Program Control Panel and Display Functions. (By Author)

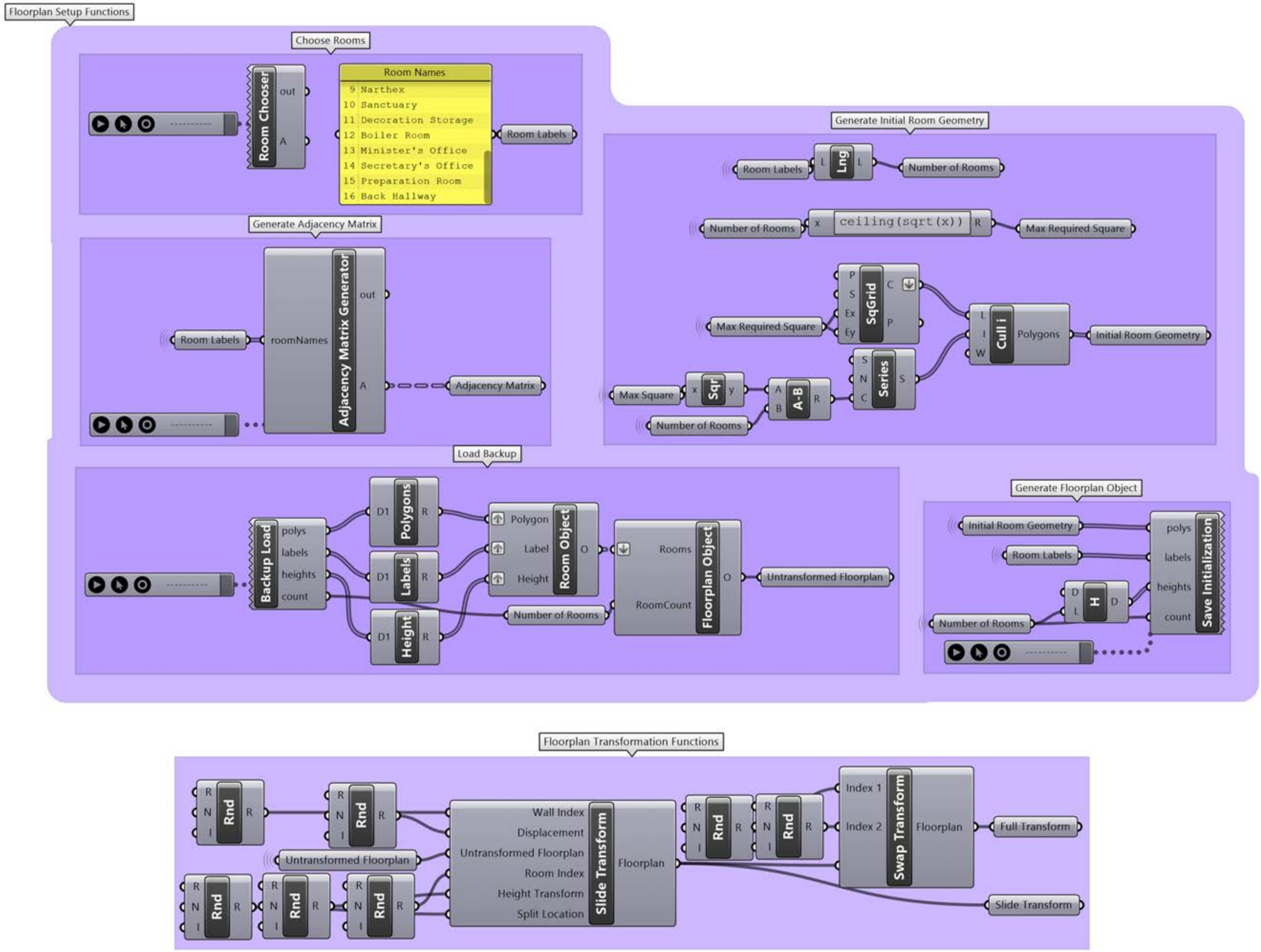


Figure 10.2 Floor Plan Setup and Transformation Functions. (By Author)

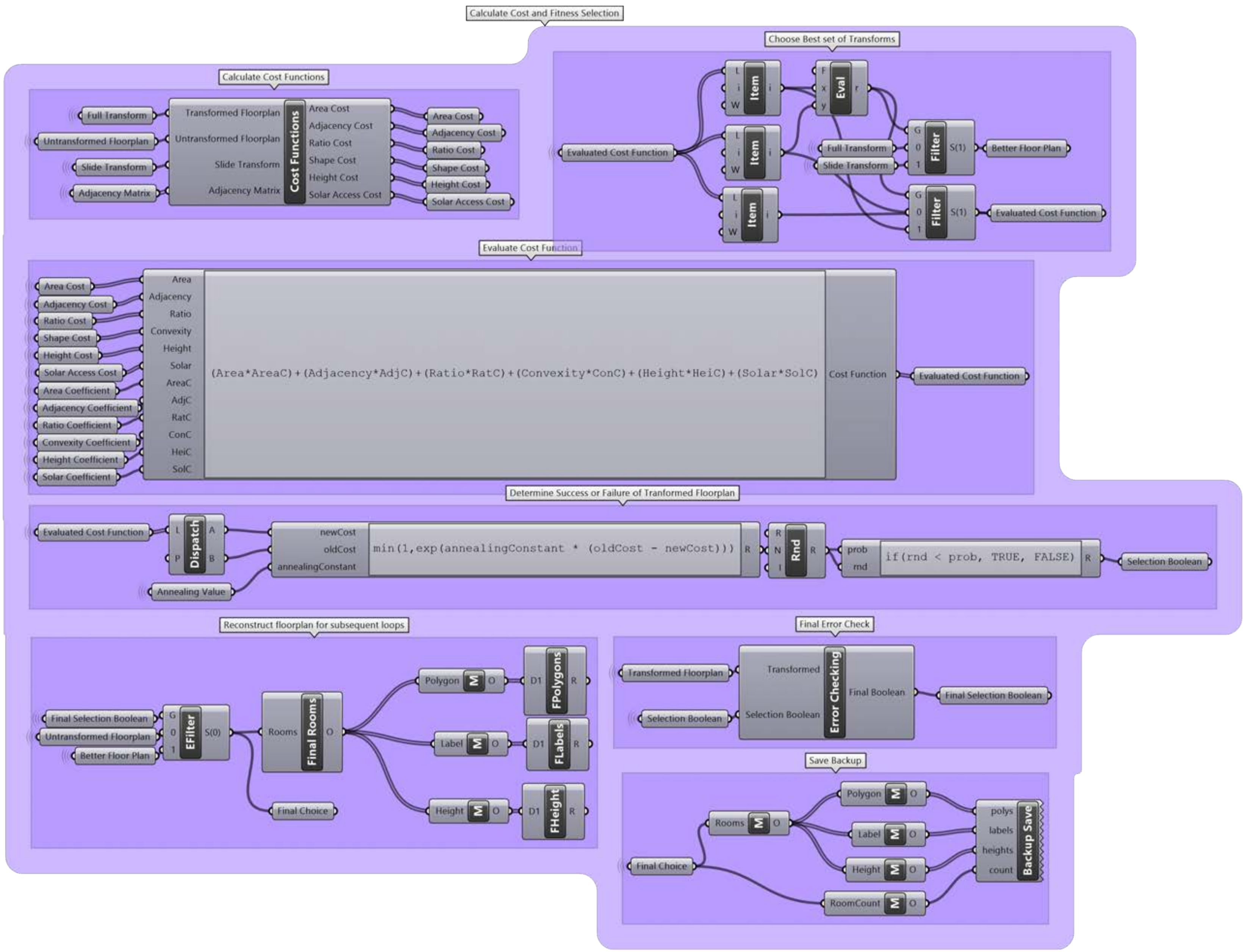


Figure 10.3 Cost Functions and Simulated Annealing Functions. (By Author)

Slide Transform

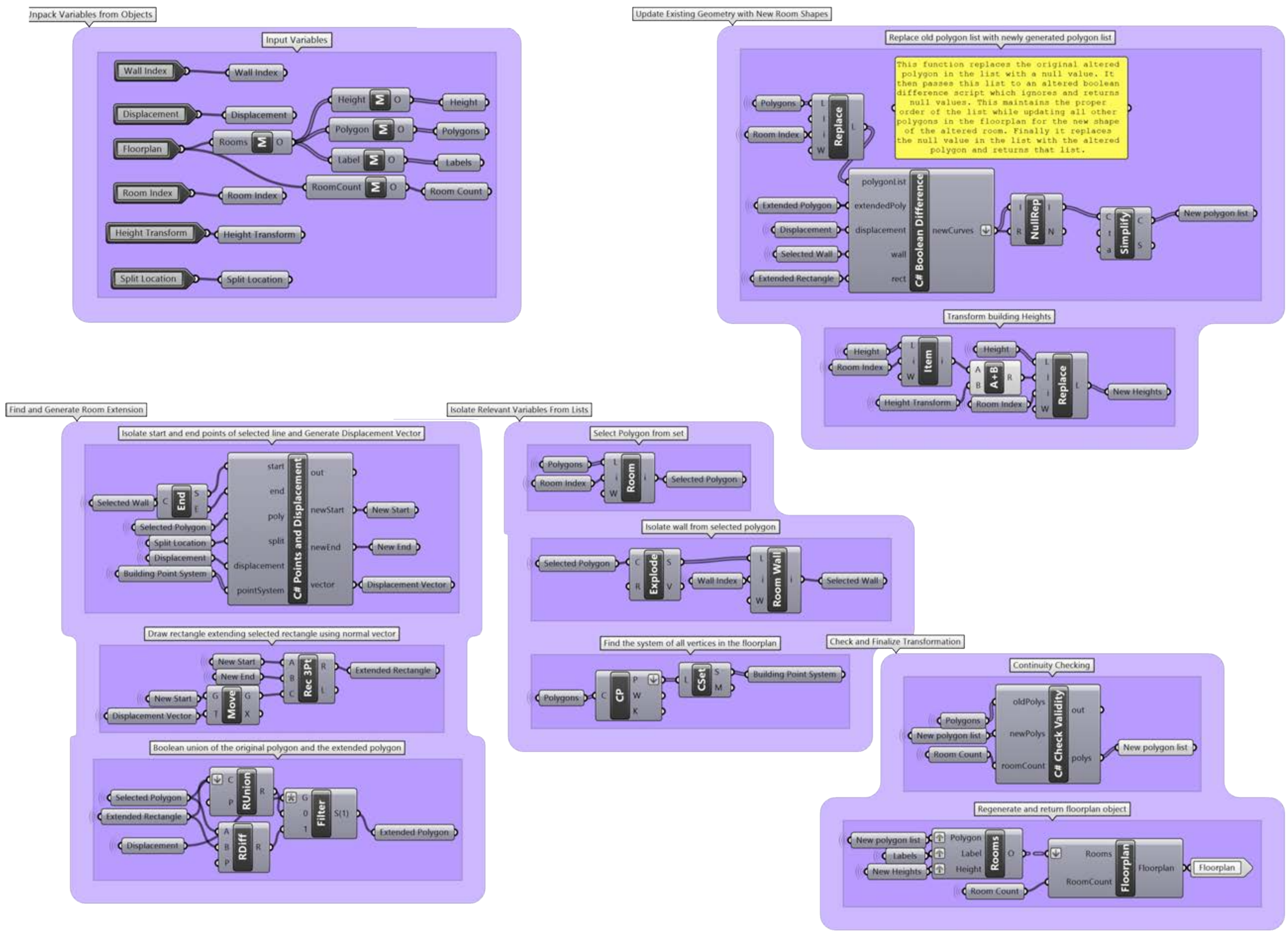
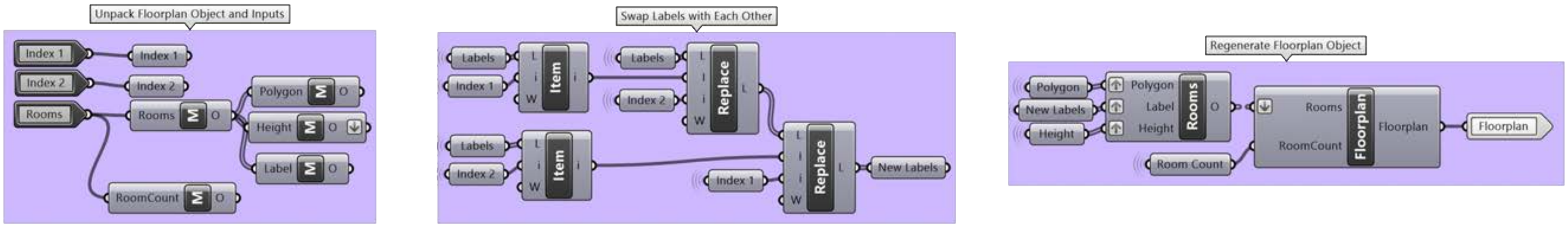


Figure 10.4 Inside the Wall Slide Transform. (By Author)

Swap Transform



Cost Functions

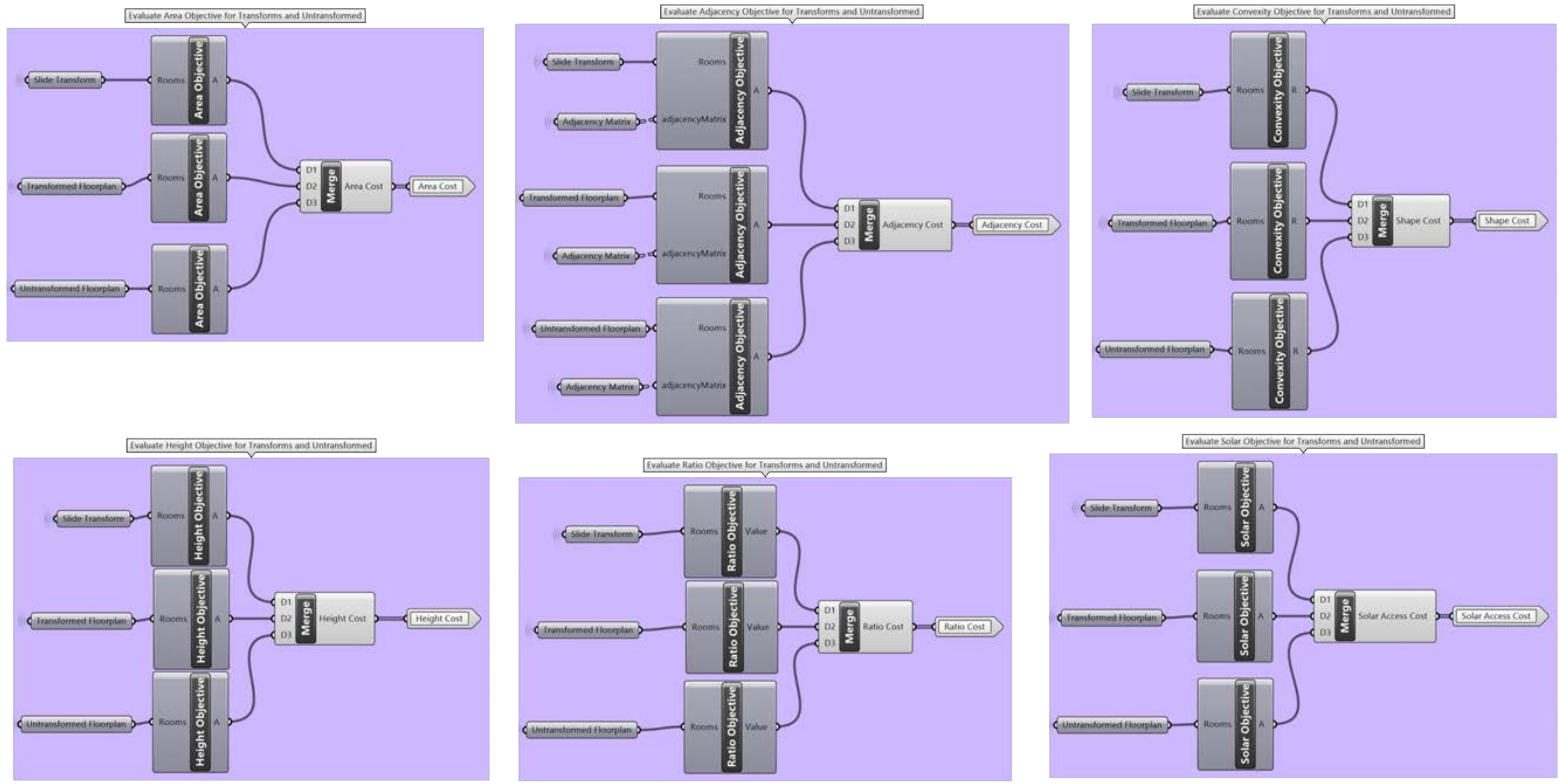


Figure 10.5 Inside the Label Swapping Transform and Cost Function Component.
(By Author)

Height and Area Cost Functions

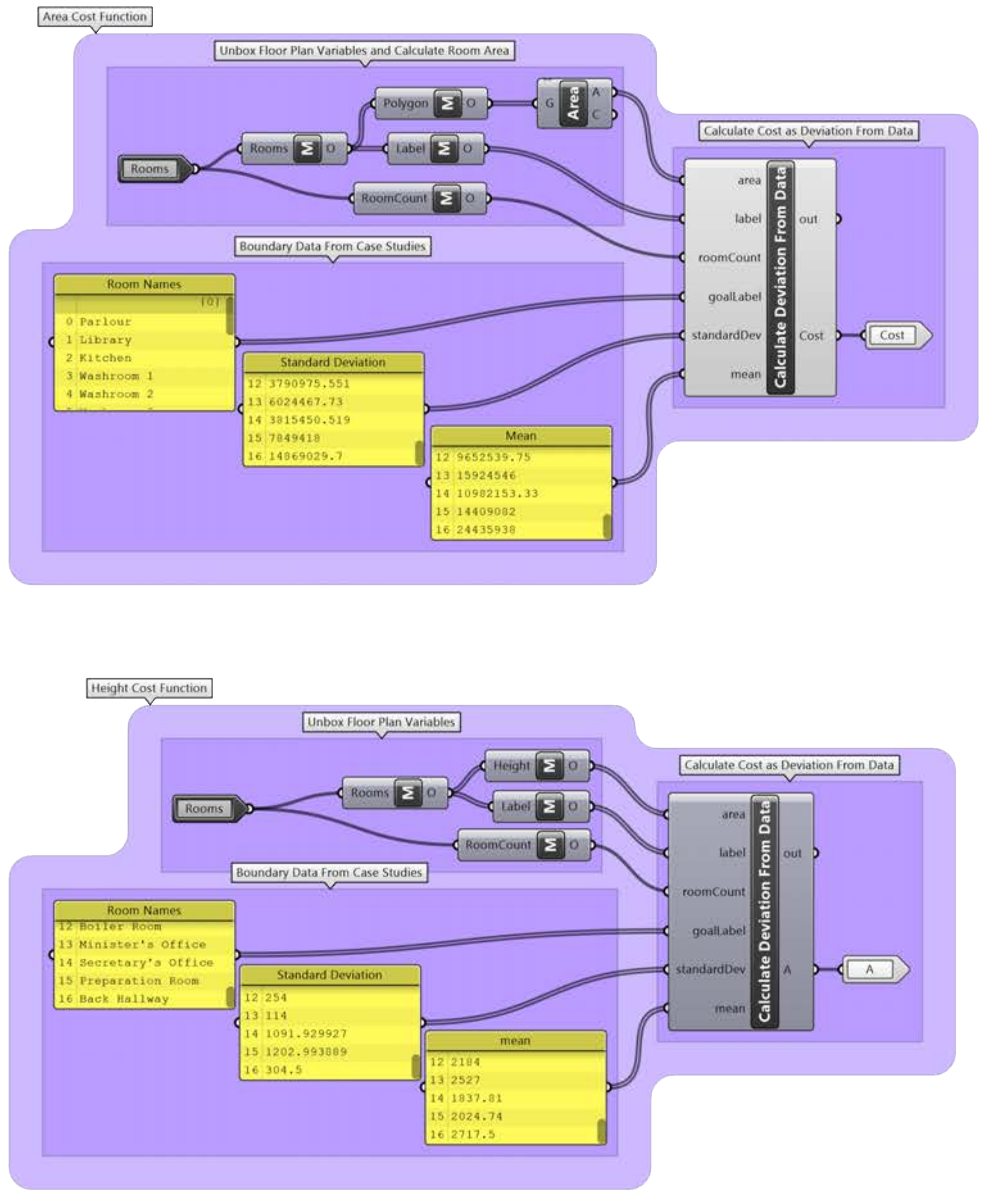


Figure 10.6 Inside the Height and Area Cost Components. (By Author)

Adjacency and Ratio Cost Functions

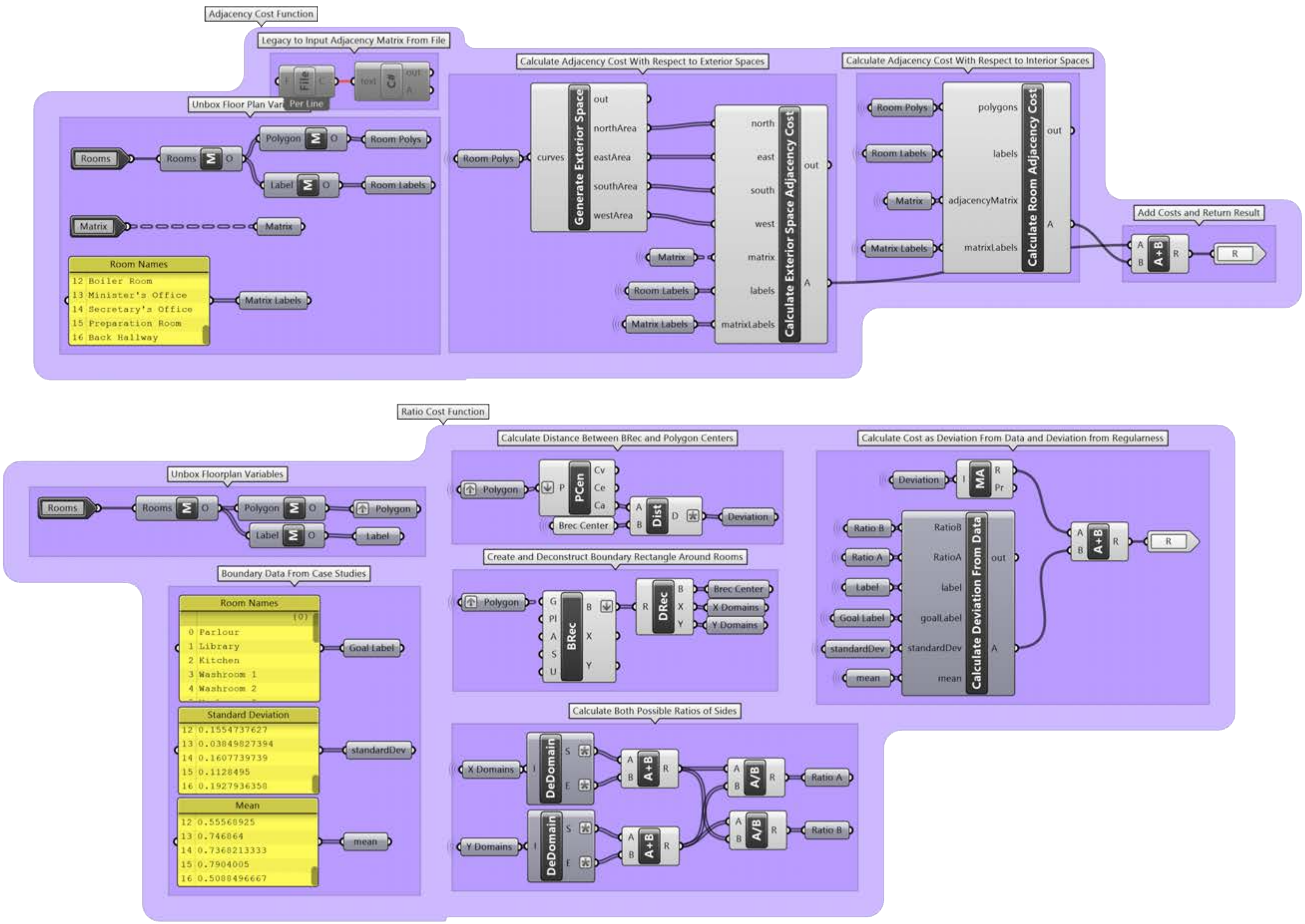


Figure 10.7 Inside the Adjacency and Ratio Cost Components. (By Author)

Shape and Solar Cost Functions

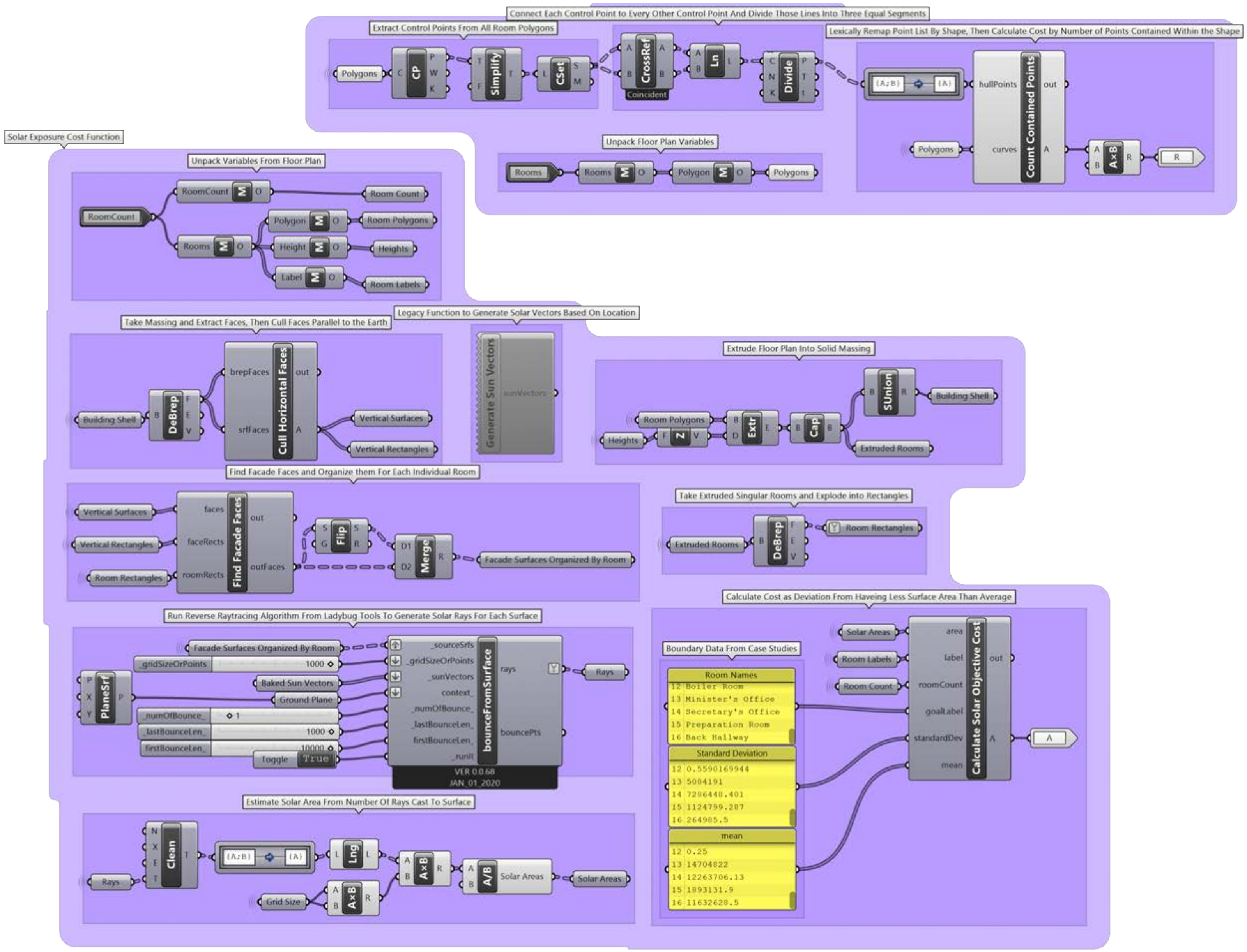


Figure 10.8 Inside the Shape and Solar Access Cost Components. (By Author)

Using The Generative System

The process of using the grasshopper definition outlined above is fairly simple. Activating the loop component on the control panel will start the simulated annealing process. The algorithm will take a floor plan loaded in from the set up functions and iteratively apply transformations to it. The setup for the plans is handled with some custom code with the setup functions. Those functions are responsible for creating a list of the names of spaces in the plan, generating a 2D boolean matrix based off of that list for the adjacency cost function, and initializing all variables related to room height and footprint.

After the initialization of and transformation of the plan, the cost functions will then determine to what degree the transformations change the given floor plan into one which matches the data set. Each cost function has this data provided to it in the form of a panel containing the optimal range of values for each space. To change the goals of the optimization, one must simply change the data supplied to the cost functions.

Three schematic designs are considered. One before transformation, one after the wall slide transformation, and one after both transformations. From this point, the design with the lowest cost has the greatest chance of being selected to go on to another round. The process as it currently exists is fairly slow and manual.

It is possible to watch the development of the design in real time, therefore the need for developing tools to alter hyper-parameters automatically (other than the annealing value which should steadily increase over time) was deemed unnecessary. Tweaking the coefficients, like most forms of hyper-parameter tuning, is therefore a subjective and inexact science. The coefficients exist more or less to allow the user some control over the extent to which the optimization process favors any particular attribute. In use, if any biases were noticed as leading to unfavorable outcomes, that parameter was tuned down to allow for others to manifest. If any attributes were seen as lacking, those parameters were turned up. In general, the process kept all coefficients on the cost functions equal except in the prior cases.

Should the user want to stop the process, they can halt the loop component and save the plan to a file directly from the definition. In the event that the user should need to start the process up again, it is also possible to load a plan from file and re-engage the loop component.

Works Cited

(Figure 1.5) Hillier, Bill, and Julienne Hanson. “The Small Town of G” from *The Social Logic of Space*. Cambridge University Press, 1984. Figure 26,27,28.

(Figure 2.1) Al-Jokhadar, Amer, and Jabi Wassim. “Humanizing The Computational Design Process.” In *Parametricism VS. Materialism Evolution of Digital Technologies for Development*. London: Imperial House Publishers, 2016. Figure 3.

(Figure 2.2) Merrell, Paul, Eric Schkufza, and Valdlen Koltun. “Computer-Generated Residential Building Layouts.” *ACM Transactions on Graphics*, December 2010. https://doi.org/10.1145/1866158.1866203. Figure 5.

(Figure 2.3) As, Imdat, Siddharth Pal, and Prithwish Basu. “Artificial Intelligence in Architecture: Generating Conceptual Design Via Deep Learning.” *International Journal of Architectural Computing 16* (2018): 306-27. Figure 3.

(Figure 4.1) Rathwell, Natalie Anderson. DOMINION-CHALMERS, LOOKING NORTH ON O’CONNOR STREET. In Heritage Value in *Ottawa’s Dominion-Chalmers United Church: History*, *Community*, *Sight*, *and Sound*. 1st ed. Vol. 43. 47-56. Accessed January 10, 2021. Erudit.

(Figure 4.2) Rathwell, Natalie Anderson. VIEW OF THE DOME FROM BEHIND THE PULPIT, DOMINION-CHALMERS. In Heritage Value in *Ottawa’s Dominion-Chalmers United Church: History*, *Community*, *Sight*, *and Sound*. 1st ed. Vol. 43. 47-56. Accessed January 10, 2021. Erudit.

Alexander, Christopher. *The Timeless Way of Building*. Oxford University Press, 1979.

Alexander, Christopher, Sara Tshikawa, and Murray Silverstein. *A Pattern Language*. Oxford University Press, 1977.

Al-Jokhadar, Amer, and Jabi Wassim. “Humanizing The Computational Design Process.” In *Parametricism VS. Materialism Evolution of Digital Technologies for Development*. London: Imperial House Publishers, 2016.

As, Imdat, Siddharth Pal, and Prithwish Basu. “Artificial Intelligence in Architecture: Generating Conceptual Design Via Deep Learning.” In *International Journal of Architectural Computing 16 (2018)*: 306-27.

Bidgoli, Ardavan, and Pedro Veloso. “DeepCloud: The Application of a Data-Driven, Generative Model in Design.” In *Recalibration on Imprecision and Infidelity*. 186, 2018.

Canada. Statistics Canada. Religions in Canada. May 13, 2003. Accessed January 7, 2021. https://www12.statcan.gc.ca/english/census01/Products/Analytic/companion/rel/canada.cfm.

Chaillou, Sanislas. “AI + Architecture: Towards a New Approach.” *Harvard University Graduate School of Design*, 2019.

“Community History.” The Onaping Falls News. Accessed August 15, 2021. http://www.onapingfallsnews.com/community-history/.

Gibson, James. *The Ecological Approach to Visual Perception*. Psychology Press, 2014.

Harman, Graham. *The Quadruple Object*. John Hunt Publishing, 2011.

Heimstra, Rick, and Karen Stiller. “Religious Affiliation and Attendance in Canada.” *In Trust*. Accessed January 5, 2021. https://www.intrust.org/Magazine/Issues/New-Year-2016/Religious-affiliation-and-attendance-in-Canada.

Hillier, Bill. *Space Is the Machine*. University of Cambridge, 2007. https://spaceisthemachine.com/.

Hillier, Bill, and Julienne Hanson. *The Social Logic of Space*. Cambrige University Press, 1984.

Levine, Caroline. *Forms: Whole*, *Rhythm*, *Hierarchy*, *Network*. Princeton University Press, 2015.

Merrell, Paul, Eric Schkufza, and Valdlen Koltun. “Computer-Generated Residential Building Layouts.” *ACM Transactions on Graphics*. December 2010. https://doi.org/10.1145/1866158.1866203.

Mitchell, William. *The Logic of Architecture: Design, Computation, and Cognition*. MIT Press, 1990.

Monchaux, Nicholas de. Spacesuit: Fashioning Apollo. *MIT Press Cambridge*, Massachusetts, 2011.

Morton, Timothy. *Realist Magic: Objects, Ontology, Causality*. n.d.

Pask, Gordon. “The Architectural Relevance of Cybernetics.” *Architectural Design*, no. 7/6 (1969) 494-96.

Rathwell, Natalie Anderson. “Heritage Value in Ottawa’s Dominion-Chalmers United Church: History, Community, Sight, and Sound.” *Journal of the Society for the Study of Architecture in Canada* 43, no. 1 (January 10, 2018): 47-56. Accessed January 10, 2018. doi:https://doi.org/10.7202/1049407ar.

Scott, David. *Gilbert Simondon’s Psychic and Collective Individuation*. Edinburgh Universigh Press, 2014.

Stamper, Ronald. *A Semiotic Theory of Information and Information Systems*. University of Twente, 1993. https://ris.utwente.nl/ws/portalfiles/portal/5383733/101.pdf.

Steenson, Molly. *Architectural Intelligence: How Designers and Architects Created the Digital Landscape*. The MIT Press, 2017.

Vaughan, Mitchell. “The Capability-affordance Model: A Method for Analysis and Modelling of Capabilities and Affordances.” In *Proceedings of the Second International Symposium on Business Modeling and Software Design*, 60-71. Science and Technology Publications, 2012. Accessed November 13, 2020. https://www.scitepress.org/papers/2012/44612/44612.pdf.

Williams, Mani, Jane Burry, and Asha Rao. “Understanding Social Behaviors in the Indoor Environment: A Complex Network Approach.” In Proceedings of the *34th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA)*, 671-80. Netherlands, 2014. https://researchbank.rmit.edu.au/view/rmit:28597.

Yearbook and Directory: United Church of Canada. Toronto, Ontario: United Church of Canada, 2019.