Architectural Epidemiology
A Computational Framework

by Jim Peraino

M. Arch I Harvard University Graduate School of Design (2015)
B.A. Washington University in St. Louis (2010)

Submitted to the Department of Architecture and
the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degrees of

Master of Science in Architecture Studies and
Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2020

Author ____

Department of Architecture
Department of Electrical Engineering and Computer Science
May 8, 2020

Certified by ____

Takehiko Nagakura
Associate Professor of Architecture
Thesis Supervisor

Accepted by ____

Leslie K. Norford
Professor of Building Technology
Chair, Department Committee on Graduate Students

Accepted by ____

Leslie A. Kolodziejski
Professor of Electrical Engineering and Computer Science
Chair, Department Committee on Graduate Students

Thesis Committee

Takehiko Nagakura
Associate Professor of Architecture
Thesis Advisor

Michael Stonebraker
Adjunct Professor of Electrical Engineering and Computer Science
Thesis Reader

Andrea Chegut
MIT Real Estate Innovation Lab Director
Thesis Reader

Architectural Epidemiology
A Computational Framework

by Jim Peraino

Submitted to the Department of Architecture and
the Department of Electrical Engineering and Computer Science
on May, 8, 2020 in Partial Fulfillment of the Requirements for the Degrees of

Master of Science in Architecture Studies and
Master of Science in Electrical Engineering and Computer Science

Abstract

Architecture affects our health, especially in hospitals. However, our ability to learn from existing hospitals to design buildings that improve patient outcomes is limited. If we want to leverage large datasets of health outcomes to build knowledge about how architecture affects health, then we need new methods for analyzing spatial data and health data jointly. In this thesis, I present several steps toward the goal of developing a computational model of architectural epidemiology that aims to leverage both human and machine intelligence to do so.

First, I outline the need for structured architectural datasets that capture spatial information in schemas that current drawing formats do not allow. These datasets need to be wide to capture multifaceted and qualitative aspects of the built environment, and so we need new methods to generate this data. Finally, we need strategies for surfacing insight from these datasets by involving both humans and machines in the process.

Next, I propose a framework to satisfy these criteria that consists of four components: 1) data sources, 2) feature engineering, 3) statistical analyses, and 4) decision-making activities. Two case studies provide in-depth illustrations of these components: The first presents a 3D interface that enables developers to create 3D visualizations of large health outcome datasets in architectural space while taking advantage of the Kyrix details-on-demand system's backend performance optimizations. The second tests the efficacy of neural network ablation to surface relationships between architectural characteristics and health outcomes using a synthetic dataset.

It is not necessary to ignore human intuition if we want to take advantage of computational power, and it is not necessary to leave behind computational power if we want to take advantage of human intuition. By overcoming current technical barriers with the methods proposed in this thesis, we can work toward achieving both. Ultimately, we can learn from our current environments to design buildings that improve our health.

Thesis Advisor
Takehiko Nagakura
Title: Associate Professor of Architecture

Acknowledgments

Takehiko Nagakura
I'm grateful for your thoughtful guidance and feedback over the past two years.

Michael Stonebraker, Wenbo Tao, El Kindi Rezig, Eirca Zhao, and the CSAIL Data Systems Group
Thank you for the chance to play a small part in a big project, and for the help along the way.

Andrea Chegut and the MIT Real Estate Innovation Lab
Thank you for building a community dedicated to measuring the value of design, and for inviting me in.

Sonal Singh
Thank you for working tirelessly to turn ideas into reality.

Alan Ricks, Michael Murphy, Regina Yang, Patricia Gruits, Amie Shao, Jeff Mansfield, and everyone at MASS Design Group,
I'm grateful for the time I spent learning from you. You taught me that architecture matters.

Mom, Dad, Drew
You've made so much possible, in so many ways. Thank you.

Contents

1. Introduction

Architects face difficult choices during the design process, and we make them without being able to take full advantage of the evidence at our disposal. We are constrained–by budgets, by sites, by geometry, and as a result, we must make trade-offs. This is especially the case in hospitals, where design decisions can mean the difference between life and death.

A growing body of evidence demonstrates that architectural characteristics in hospitals such as how visible a patient is from a nurse station can affect health outcomes like mortality rates.[21] Some studies claim that patients in rooms with views to nature may recover faster and request less pain medication than patients with views to a brick wall.[37] We do not yet have a full understanding of these relationships and when they hold, but the pattern is clear: architecture affects our health, and we have a duty to make design decisions that take this into account.

For an architect that sets out to do so, there is remarkably little support as they make design trade-offs. Architects put forth design principles that guide these decisions and cite studies to back them. An architect may understand that fitting an additional patient room into a layout will increase the numbers of patients that a hospital can serve, and that including a lounge will reduce stress and allow staff to provide better care to their patients, but what happens when there is not enough room for both? One might propose a cost-benefit analysis: consider how many life-years an additional patient room would save through increased capacity versus the number of life-years that less overburdened staff would save, and choose accordingly. But that kind of analysis is not currently possible–no such data is available to architects during the design process.

As other industries build large datasets to enable data-backed decision-making, architects remain largely unable to take advantage of the lived experiences that have transpired in millions of buildings across the world to design better buildings. If we want to do so, then we need frameworks for understanding how architecture affects our health and computational methods for implementing them.

In this thesis, I take several steps toward the goal of leveraging architectural evidence to improve future designs by interrogating the reasons we have not yet progressed and outline several methods for overcoming these hurdles. In doing so, I build on related efforts to propose a computational framework for *architectural epidemiology*, or the study of how design affects our health.

In **Chapter 2**, I provide context for this framework, identifying opportunities and barriers to implementation. Nurses, physicians, designers, and epidemiologists have been working to understand relationships between our physical environment and our health for the past two centuries. Their efforts demonstrate that drawing conclusions about and acting upon these relationships is important yet complicated; efforts rarely result in conclusive results nor in design heuristics that architects can deploy universally. Several barriers contribute to this problem: First, architecture affects us in indirect and interdependent ways; influences can be challenging to untangle. Second, we lack large, structured architectural datasets that are rich enough to capture the aspects of our environments that affect our health; without access to evidence, drawing conclusions is not possible. Third, neither human nor machine intelligence is well-suited to tackle this problem alone; we need methods to leverage humans' ability to validate data, frame problems, and account for factors that are difficult to capture in data. We also need systems to leverage computation to navigate massive datasets, recognize patterns, and conduct analyses that would take us too long to do manually. A framework for *architectural epidemiology* must, therefore, make it easy for humans to augment machines' efforts, and for machines to augment humans' efforts.

In **Chapter 3**, I propose a **framework for *architectural epidemiology*** that aims to satisfy the criteria established in the previous chapter. Data science and machine learning techniques for recognizing patterns and predicting outcomes are well-established; my emphasis here is on highlighting domain-specific considerations. To that end, I first highlight several potential datasets and propose avenues for overcoming obstacles that limit their use in practice. Next, I identify processes for translating qualitative spatial characteristics into quantitative datasets that can serve as inputs for data science and machine learning models. Then, I weigh the merits of several data science and machine learning methods, discussing how researchers can deploy them for various design analysis tasks. Finally, I identify techniques and activities that can be deployed during design and analysis processes to take advantage of both human and machine intelligence to inform design processes.

I present two case studies that provide a more tangible illustration of the challenges that the framework needs to overcome. These more in-depth studies were selected to consider opposite ends of the human-machine interaction spectrum.

In **Chapter 4**, I present a new **3D data visualization and discovery frontend** that enables users to navigate electronic medical record data in a 3D model of a hospital campus. This system aims to harness human intuition in the data validation and discovery process. It highlights several challenges in integrating spatial data into the design process; architectural drawings are often unstructured and nonstandard. I propose a method for associating room names and levels with geometric objects to generate structured datasets. Working with massive datasets like electronic medical records can limit performance and make real-time interaction difficult. This implementation builds on the Kyrix details-on-demand system developed by the Data Systems Group at MIT's Computer Science and Artificial Intelligence Lab (CSAIL) to leverage its backend optimizations, making fluid interactions possible.

In **Chapter 5**, I present a case study using synthetic data and a neural network ablation analysis to evaluate the extent to which different spatial characteristics can predict an outcome variable such as patient mortality rate or length of stay. In contrast to the 3D visualization case study, the goal of this study is to leverage machines' ability to comb through large amounts of data to surface trends. The case study emphasizes how architecture can serve as an input to a neural network via a process of feature engineering.

Finally, I reflect on challenges and next steps for this research in **Chapter 6**.

The work presented in this thesis does not claim to be comprehensive nor to solve the problem of optimizing buildings for health outcomes with an end-to-end solution. Instead, my goal is to build upon established domains of evidence-based design, space syntax, and machine learning to demonstrate that although no perfect solution may exist, we can do much better than the status quo. It is not necessary to ignore human intuition if we want to take advantage of computational power, and it is not necessary to leave behind computational power if we want to take advantage of human intuition. By overcoming current technical barriers with the methods discussed and proposed in this thesis, we can work toward achieving both. Ultimately, we can learn from our current buildings to design buildings that improve our health.

2. Background

Architects, planners, and epidemiologists have deployed wide-ranging techniques to understand the mechanisms by which architecture affects our health. The past two hundred years, in particular, have seen new building typologies devoted to healing, new types of data visualizations that enable new disciplines of epidemiology, and new methodologies for researching and codifying knowledge about the built environment. If we can learn about the ways that our buildings influence our health, then we can wield this understanding to improve public health.

Any computational approach to this goal should learn from the opportunities and limitations that current and previous efforts have elucidated. To that end, this chapter provides an overview of this lineage over the past two centuries with the goal of establishing a set of criteria that a computational approach of *architectural epidemiology* should satisfy.

2.1 Sites and contexts

By the mid-nineteenth century, public health had become a key consideration in urban design and planning. Frederick Law Olmstead's plan for Boston's Back Bay Fens, for example, targeted health issues caused by tidal flats that had been overrun with sewage. By developing the area into a healthy ecosystem and mitigating public health concerns, he turned a previously undesirable area at the periphery of the city into an appealing location for new residents.[20] In 1861 Olmstead was appointed the director of the U.S. Sanitary Commission, essentially working in the capacity of a public health official for the Union Army during the U.S. Civil War to oversee camp sanitation for volunteer soldiers.[17]

Around the same time, physicians began opening sanatoria, facilities for the treatment of tuberculosis. Often located in the countryside so that patients could be exposed to fresh air that was missing from the cities, these facilities were the preferred treatment for the illness prior to antibiotics. Previously, patients opted to be treated at home; healthcare facilities were often considered places where disease spread rather than was cured. Now, the built and natural environment was prescribed and used as a treatment in itself.

2.2 Uncovering environmental determinants of health with data visualization

John Snow and the Broad Street cholera outbreak (1854)

These new ways of thinking about the relationship between our environments and our health required new modes of representation. Diagrams became tools of both explication and communication. When the epidemiologist John Snow combined

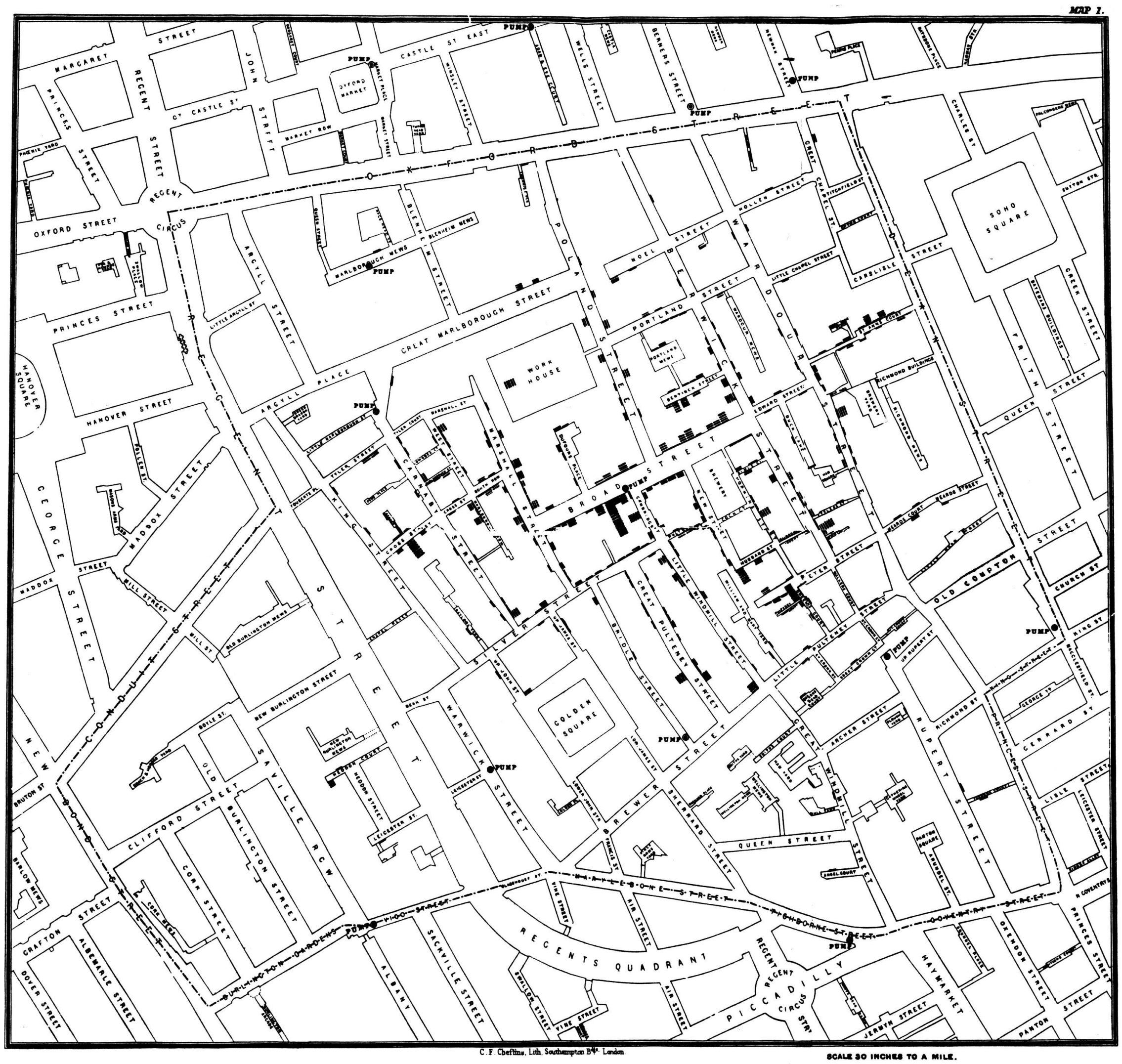

Figure 2-1: John Snow's map tracking the locations of illnesses during a cholera outbreak illustrates the potential of data visualization to diagnose the source of disease.[30]

medical data with spatial data in the mid 19th century, he discovered the source of an intractable cholera outbreak and upended conventional wisdom of how disease spread through the city. Prior to his study, which consisted of mapping the locations of sick patients as an overlay to a street map, doctors suspected that Cholera was an airborne illness, and prescribed precautions accordingly.[28] With Snow's new insight at hand, officials could remove the well's handle to prevent use and stymie the spread of the illness.

Florence Nightingale's *On Hospital Reform* (1863)

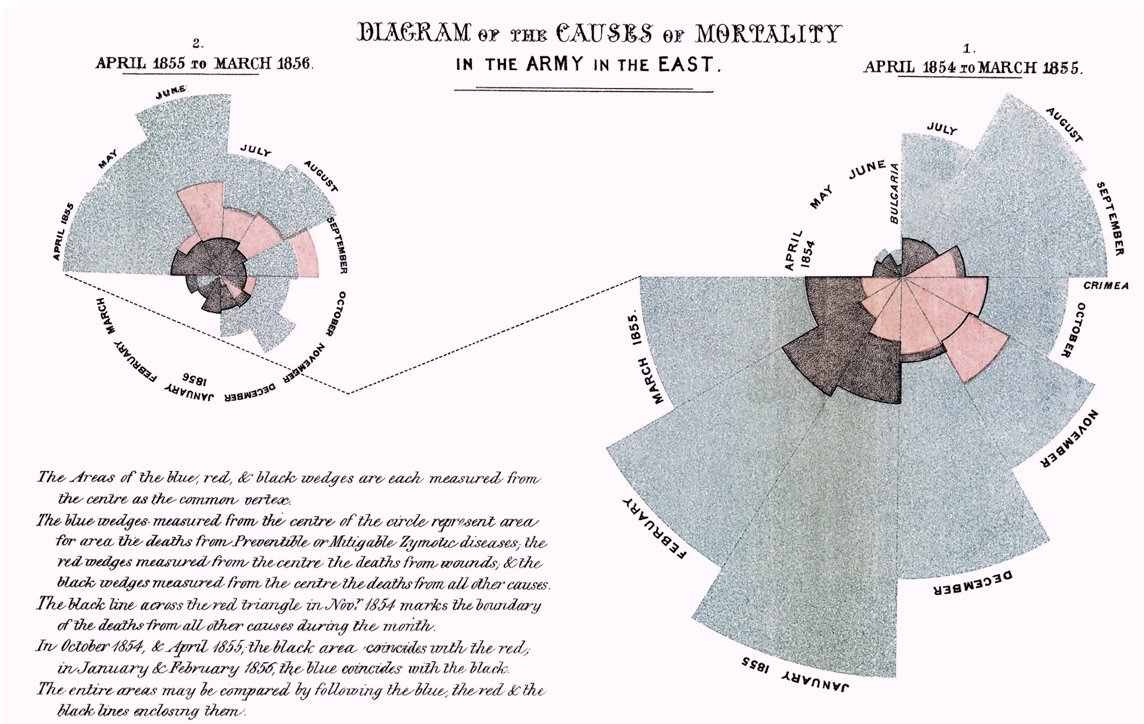

Figure 2-2: Nightingale's coxcomb charts are early examples of data visualization, and were used to make the argument that architecture was affecting health.[19]

Florence Nightingale's work in data visualization similarly unearthed trends that weren't immediately obvious. A nurse and a statistician, Nightingale was dispatched to the British Army's base at Scutari Turkey in 1853 during the Crimean war, where she described dire conditions. The facilities were dirty, poorly lit, cramped, uncomfortable, and mortality rates were as high as 42.7%.[18] Nightingale's coxcomb charts are often cited as early examples of data visualizations; she used these diagrams to illustrate that the army was suffering significantly more deaths from diseases that were rampant at the infirmaries than from actual battle wounds, thereby creating the impetus to act.

She was an early advocate for spatial data, calling for the following details to be recorded at each facility: "The number of beds. The number of storeys. The number of wards. The length, breadth, and height of wards. The number of beds per ward. The cubic feet per bed. The superficial area per bed. Number of windows, with their dimensions. Means of ventilation. Drainage. Water-closets or latrines. Water supply".[18] As a result of this new spatial data, visual analyses, and the recommendations that were informed by them, Nightingale was able to convince lawmakers to make changes that reduced the mortality rate from 42.7% to 2.2%.[18]

2.3 Incorporating evidence with design guidelines

These early forms of data visualization enabled direct action to solve urgent problems but did not yet address systemic gaps across the built environment. At the end of World War II, shifting landscapes in the United States required rethinking the network of facilities that would treat returning war veterans, accommodate mass migration to the suburbs, and take advantage of new developments in medicine. Success would depend on significant coordination and capital investment. Congress passed the Hill-Burton Hospital Construction Act of 1946 in response. The act provided funding for the planning, construction, and to some extent, standardization of facilities, ultimately providing $33.1 billion in funds over three decades, funding more than 5,000 projects.

Since construction was planned at such a large scale, there was a vested interest in ensuring that best practices were developed and followed. In response, the U.S. Public Health Service (USPHS) provided funding for research to investigate optimal designs. As hospital administrators began early phases of facility planning, many USPHS funded studies were targeted to improve hospital performance.

AHA Hospital Design Checklist (1965)

One such effort was a pamphlet entitled *Hospital Design Check List*, published by the American Hospital Association (AHA) in 1965. It featured 45 pages of architectural considerations to be evaluated during reviews of hospital floorplans. For each of the approximately 2,000 items, the reviewer is asked to indicate whether the feature is "satisfactorily provided," "desirable but not necessary," or "should be restudied" in their plans. Items range from a simple check to see whether or not components are included (nurse's supply room, oxygen control valves, portable emergency light), to performative issues (nurses' visual control, location of pharmacy with respect to access to elevators), and occasionally more subjective aspects (color scheme).[?]

Whereas Nightingale made specific claims about sizes, locations, and configurations, the AHA checklist leaves it to designers and administrators to make these decisions; no judgment is provided on the merits of any decision. Instead, the AHA argues that each facility will have different demands and that the checklist method accommodates the requirements and preferences of the facilities' architects and administrators. It argues that "this method of measuring the probable effectiveness of architectural features for a hospital has a distinct advantage over methods employing fixed general standards that do not include all situations and cannot easily be kept abreast of advances in the many phases of patient care".[?] This acknowledgment is perhaps in line with the contingent nature of design, allowing the designer and administrators to weigh the relative importance of a variety of factors. Intense studies on specific aspects of design are still possible. Still, it acknowledges a problem of multivariable optimization: optimizing for the performance of one variable often comes at the cost of the performance of another.

The AHA checklist puts the onus on the architect and administrator to balance the wide-reaching considerations.

2.4 Evidence-based design

More recent efforts to assess how architecture affects health have taken advantage of techniques like difference-in-difference analysis, natural experiments, and randomized control trials.

Ulrich's view to nature study (1984)

One oft-cited study is Roger Ulrich's 1984 investigation that found through a natural experiment that a view to nature from a patient's room as they recover could lead to shorter recovery times and lower pain medicine intake.[37] The study considered nine years of data from a ward that consistently served cholecystectomy patients. Nurses assigned patients to rooms as they became vacant, and Ulrich controlled for considerations such as a patients' preexisting conditions, history of previous hospitalization, and wall color. The goal was to isolate a single variation: some rooms had views to foliage while others had views to a brick wall. Ulrich did a remarkable job of addressing confounds. Still, he provides a warning about the generalizability of his findings: "The conclusions cannot be extended to all built views, nor to other patient groups, such as long-term patients, who may suffer from low arousal or boredom rather than from the anxiety problems typically associated with surgeries. Perhaps to a chronically under-stimulated patient, a built view such as a lively city street might be more stimulating and hence more therapeutic than many natural views".[37]

2.4.1 Evidence-based design glossary (2011)

Studies in the vein of Ulrich's view to nature study accumulated over time, and a set of architects, interior designers, and researchers founded the nonprofit organization the Center for Health Design in 1993.[?] In 2011, a team led by Ulrich conducted a literature review of hundreds of studies considering architecture's role in health outcomes.[23] Priority outcomes included health outcomes, patient satisfaction, and operational efficiency.

Studies continue to add to these findings. Researchers have investigated hypotheses that architecture can contribute to patient outcomes by reducing nosocomial infections,[41] medical errors,[5] and patient anxiety,[4] while encouraging healthy behavior change like hand washing[7] and caregiver-patient interactions.[4]

Several aspects of the built environment may influence patient satisfaction, including comfort,[14] aesthetic perception,[31] and proximity to nursing stations.[16] High-quality physical environments can positively influence perceptions of care, reduce anxiety, and foster better communication with staff.[4] These factors may also contribute to improved patient outcomes via a placebo effect.[24]

Hospital layout can affect operational metrics like staffing efficiency and team cohesion[11] while enabling higher quality communication between staff.[13] Nurses that need to spend more time traveling between patient rooms due to inefficient layouts may suffer more fatigue and spend less time with patients.[27] Light and sound at nursing stations can support or impair nurse performance.[9]

2.5 Discussion

The preceding examples provide context for the framework outlined in the next chapter. First, they motivate the approach by demonstrating that architecture affects our health but that we are just beginning to scratch the surface when it comes to harnessing these relationships. Then, they highlight several design considerations for a framework that aims to expand these capabilities.

Architecture affects our health.

Hospital architecture affects patient health outcomes, like mortality rates and pain medication intake. It affects operational outcomes such as staff burnout, team cohesion, and travel distances. It affects the patient experience. It is critical, therefore, that we learn more about these relationships and develop methods for integrating our findings into the design process.

We need to be aware of omitted-variable bias.

Architecture is never the only factor that determines a patient's outcomes; preexisting medical conditions, the care provided by their medical team, and cultural factors can play a greater role. We, therefore, need to be aware that the results of any given analysis may have limited relevance outside of its immediate context. A study of the effect of nurse supervision on patient mortality rates in an ICU will have limited generalizability to general inpatient wards, for instance.

Further, architectural characteristics are interdependent and difficult to isolate. For instance, windows provide both views and access to daylight. A study that finds a relationship between daylight and patient comfort may actually be picking up the effects of views. It is critical, therefore, to have holistic spatial datasets that capture multiple qualitative facets of architectural spaces.

We need larger spatial datasets if we want to generate insight at scale.

The studies that demonstrate the impact of our environments on our health are limited in scope, but with the advent of electronic medical records, there is the potential to significantly expand the scope of these insights. Although large repositories of patient data exist, no such repository of spatial data exists that contains the breadth and depth of data necessary to characterize the relationships we wish to study at scale while avoiding omitted-variable bias. While some spatial data such as square footages and locations are tracked, it does not capture the qualities of space that are relevant for the task at hand.

Architecture affects our health by enabling staff to have clear sightlines to patients, by providing comfortable settings for patient recovery, and by minimizing travel distances between essential services. These characteristics are not represented explicitly in architectural drawings, but instead, need to be extracted from unstructured drawings through a process of analysis. To generate structured, consistent, and rich datasets at scale, we'll need methods to standardize and automate these analyses.

We need to leverage both human and machine intelligence.

Human intuition is a powerful design tool, but will not be capable of keeping track of every factor that needs to be considered in the design process. Because of the breadth of the data necessary for these analyses and the contingent results of each study, we'll need to provide computational methods for designers to access relevant information without manually sifting through every data point and study.

At the same time, computation on its own will fall short on its own. Though generative design offers the promise that these guidelines could be codified and designs automated, the considerations of healthcare design are likely too complex and contingent on their context to be fully addressed by a generative design process. Operational subtleties such as staffing models and culture affect the way that spaces are used, and therefore the ways that a new design will be used.

The tension between these two types of intelligence has been debated hotly for decades. A reliance on computation requires the belief that design can be treated as a science and formalized into rules. Herbert Simon argues that "a science of design, a body of intellectually tough, analytic, partly formalizable, partly empirical, teachable doctrine about the design process" is possible.[29] Though some principles may be formalized, there remain aspects of the design process that prove more difficult, if impossible, to formalize. In articulating his concept of *reflective pracitce*, the philosopher Donald Schön notes that "indeterminate zones of practice–uncertainty, uniqueness, and value conflict–escape the canons of technical rationality. When a problematic situation is uncertain, technical problem solving depends on the prior construction of a well-formed problem–which is not itself a technical task".[26] The balance comes in merging that which is formalizable with that which is not.

2.5.1 Conclusions

A computational framework for architectural epidemiology has the potential to improve hospital design and improve patient health outcomes. It will require more and wider data to overcome omitted-variable bias and provide a holistic representation of architectural space. It will need to rely on computational methods for surfacing relevant insight from these larger datasets. It will need to include humans in the loop to perform data validation, make decisions about tradeoffs, and layer in unrepresented cultural and operational factors into the decision process. Such a framework is the subject of the next chapter.

3. A Computational Framework for Evidence-Based Design

In this section, I propose a framework of *architectural epidemiology*. The domains of data science, machine learning, public health, and architecture are vast; my goal is not to provide a complete, solved solution. Instead, I highlight domain-specific barriers that have prevented such a framework from being implemented and propose means by which these barriers can be addressed. No individual step in the process on its own captures the range of challenges alone; I emphasize breadth over depth to consider a full cross-section of the pipeline, starting from data collection and ending with design decision-making.

Work in evidence-based design and space syntax provides a solid foundation for this framework; here, I illustrate how that work can be extended to utilize large-scale datasets. This framework draws from parallel efforts in real estate, where researchers and practitioners have applied data science and machine learning to the problem of learning from the built environment. Here, I look to extend the applicability of those models to the unique challenges of health outcomes.

Criteria for a computational framework

The previous chapter characterized the problem of using multiple sources of data to inform the design process. Here, I highlight several criteria that a computational model of architectural epidemiology should satisfy:

1) Analyses considering the effect of architectural characteristics on health outcomes are likely to suffer from omitted-variable bias. **We need *wide datasets* that capture multifaceted and qualitative aspects of the built environment** to increase the likelihood of capturing the relevant spatial data in our analyses.

2) No such datasets exist yet at scale for hospital architecture. **We need methods for generating structured spatial data sets by mining multiple unstructured data sources.** The scale of these efforts requires automated systems to reduce bottlenecks.

3) Human intuition on its own will not enable us to take full advantage of the data; **We need computational methods that are better suited to combing through the data and surfacing patterns.**

4) Computation on its own will not have the capacity to identify and account for exogeneity in the data, nor to make complex design decisions that depend on cultural, political, and subjective factors. **We need tools for humans to support and take advantage of computational automation.**

3.0.1 Elements of the framework

To that end, I propose a model for generating wide and deep spatial data sets and methods for benefiting from both human and machine intelligence in the design process. This framework consists of four elements:

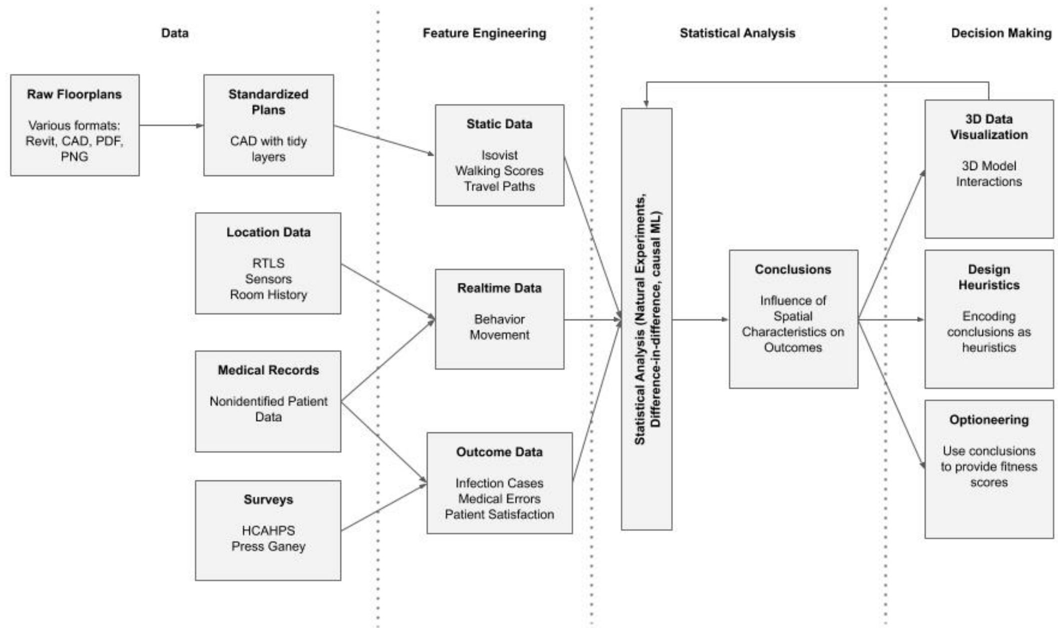

Figure 3-1: The framework consists of four components: data sources, feature engineering, statistical analyses, and decision-making.

1) **Data sources**: I identify relevant data sources and discuss their associated opportunities and limitations.

2) **Feature engineering**: I identify processes for translating qualitative spatial characteristics into quantitative data sets that can serve as inputs for data science and machine learning models.

3) **Statistical analyses**: I identify data science and machine learning techniques that are relevant to the task of answering questions data, and discuss the applicability of each approach to aspects of the task at hand.

4) **Decision making**: I identify techniques and activities that can be deployed during the design and analysis process to take advantage of both human and machine intelligence to inform design processes.

3.1 Data sources

While no cohesive architectural dataset yet exists for hospitals, several structured and unstructured data sources can be used to build one. These sources can help to create **static data** characterizing aspects of the built environment that remain constant, **real-time data** that can track inhabitants' behavior and movement, and **health outcome data** that can be used to assess the ultimate performance of a facility.

Researchers have recognized the necessity for *wide data* in applying data science techniques to research on the built environment. The MIT Real Estate Innovation Lab, led by Dr. Andrea Chegut, has research efforts specifically devoted to drawing from multiple sources to construct wide datasets. This data helps researchers assess the value of design, accounting for factors such as lease comps, building certifications, and property details.[6] Commercial solutions like Compstak and Cherre have emerged to provide data to real estate brokers to enable better investment decisions.[8] These efforts demonstrate that building up large datasets is possible but have not yet been extended to hospitals or to include characteristics that affect health outcomes.

3.1.1 Architectural data

Data containing information about buildings often comes in the form of architectural drawings or Building Information Models (BIM). These data types are ubiquitous within the architecture industry but typically exist in unstructured formats that make them ill-suited for data science applications without pre-processing. These drawings can exist in several forms: hand sketches, hand-drafted drawings, CAD files, or BIM, to name a few.

Architectural drawings are dispersed across many sources.

Drawings are often created by an architect as part of the design process and are used to communicate design intent to clients, engineers, and those tasked with constructing the building. After construction, they may be retained by the architect and building owner, distributed via publications, or used as marketing material for prospective tenants. The result is that this information may be dispersed rather than stored in a central repository.

Architectural drawings tend to highlight building components but do so implicitly.

Floor plans contain both explicit and implicit types of information. One of the primary roles of architectural drawings is to direct the construction of a particular design; to that end, they tend to contain information about building components such as walls, windows, and doors rather than emergent spatial qualities that these elements produce. While BIM models may represent these elements explicitly as components that contain associated metadata such as materials or manufacturers, they may also be represented implicitly by lines or outlines, as is the case in many DWG files or hand sketches.

Representations of qualitative aspects of design in floor plans are implicit and inconsistent.

While drawing techniques such as diagrams or renderings can be used to highlight and communicate design characteristics such as relationships between rooms, the character of a space, or the views outside of a window, these characteristics are rarely represented explicitly. Instead, they are implied by line weights, diagrammatic overlays, precedent, or convention, and in inconsistent ways across floor plans.

Architectural data requires structuring

To summarize, floor plans and BIM models are rich sources of architectural information, but access to this information is restricted due to a lack of central data warehouses, highly inconsistent formats, and a lack of explicit encoding of relevant architectural characteristics. To take advantage of the fullest extent of this information, we need to analyze plans for qualitative characteristics. To run these analyses, we need methods for extracting consistent design elements such as rooms and walls, which may be represented explicitly.

3.1.2 Sensor data

Sensor data can provide real-time insight into the activities that take place inside of a hospital, tracking how people and equipment move throughout a space. This data can be used as a process indicator. For example, if a designer is trying to understand whether a staff lounge affects burnout rates, then they need to discern whether or not staff uses the lounge since they are likely to only benefit from it if they use it. Utilization data for these lounges, as captured by sensor data, can validate this assumption.

Movement traces

Tracking and tracing movement in a space can provide details about utilization, traffic patterns, and how people socialize. **Real-time location systems** (RTLS) are one potential source of this kind of data, and can be used to track people or equipment as they move throughout a space.[10] Often implemented in hospitals to support day-to-day operations, the data generated can be used in analyses to track the effectiveness of interventions. Finer-grain locations can be tracked using equipment like the commercially available Kinect system. This kind of data can track movement and gestures, as described by Paloma Gonzalez Rojas in her thesis *Space and Motion*.[25]

Affect recognition

Real-time tracking of human affect and emotion can be achieved by using image recognition to process facial cues or wearable sensors to capture electrodermal activity. The Affective Computing Group at MIT has pioneered several related methodologies, including one study that tracked participants' skin conductance, heart rate, and self-reported mood over moth-long periods of time.[35]

Environmental qualities

Sensors deployed in buildings may also collect data related to human comfort such as temperature, humidity, ventilation, and light levels.[38]

3.1.3 Medical records

Medical records provide the primary source for outcome variables. Electronic medical records have increased in prevalence over the past decade after the Affordable Care Act of 2010 provided incentives for adopting the systems. These records contain information about a patients' medical history, treatment plans, and events such as tests, consultations, and administration of medicine. Additionally, they may include outcomes such as mortality rates and readmission rates. This data may or may not include details about the locations where the events occurred.

3.1.4 Surveys

Direct feedback from patients and staff can come in the form of digital or print forms, interviews, focus groups, or feedback terminals. Responses can be used as outcome metrics on their own, or they can shed light on model assumptions by serving as process indicators. For instance, if researchers are interested in learning about how architecture affects patient satisfaction, they may use an overall satisfaction score as an outcome variable. Several feedback terminals could also be deployed in different rooms to assess localized environmental qualities to better understand how each space contributes to the overall effect.

3.2 Feature engineering

The process of feature engineering, that of extracting data attributes from unstructured data, poses unique challenges in architectural epidemiology. In this section, I discuss several means of translating unstructured architectural drawings into numeric architectural features. I provide additional examples of feature engineering in Chapter 5, demonstrating how architectural characteristics can be transformed into inputs for a neural network.

Encodings in architectural drawings typically capture distinct elements in a space rather than implicitly encoding the resultant qualities of a space. Because of this limitation, translating floorplans into numerical features that can be input into regressions or machine learning models requires analysis. In the case of a hospital, the patient room makes a suitable unit of analysis. For each room, calculations such as the travel distance to the nearest nursing station can be input to models as numerical variables. Categorical variables, such as the view outside of a patient's room as in Ulrich's study, can also be used as inputs by one-hot-encoding.

These encodings can be straightforward, or more in-depth analysis can be conducted to quantify aspects that are typically discussed in qualitative terms. The discipline of *Space Syntax* provides many methods with which to do so, quantifying characteristics such as visibility, proximity to circulation, and connectivity.[15] Metrics like these have been used in studies that find significant results. One study proposed a new metric called *isovist connectivity* that is calculated from any given point in a plan by finding "the area of the visual polygon that is visible from anywhere within the isovist of the point".[21] Ossmann et al. found that this metric was able to predict mortality rates in ICUs.

We can use these encodings for large-scale data analyses across multiple facilities, but first, we'll need to develop methods for automating these analyses. The quality of these encodings will only be as good as the quality of the drawings that are analyzed. Not only do the analyses have to be performed in consistent ways, but the drawing elements that serve as the basis such as walls, doors, and windows, need to be accurately and consistently captured as well.

3.3 Statistical analysis

With consistent, qualitative datasets, we can leverage computation to analyze trends and surface insights. In this section, I provide a high-level overview of analysis techniques and their potential relevance to architectural epidemiology. In Chapter 6, I present a more in-depth case study assessing the viability of using neural network ablation in statistical analysis.

3.3.1 Influence weighting

Econometric methods such as **linear regression** can reveal correlations between input architectural features and output health outcomes. This is the method employed by many studies in the evidence-based design literature. However, satisfying conclusions are often elusive due to small sample sizes or potential omitted-variable bias.[23]

Natural experiments can be sought out in the built environment to strengthen conclusions, as was the case in Ulrich's landmark 1984 study, in which patients were randomly assigned to rooms that had naturally occurring variation; rooms on one side of the hall had views to nature, while rooms on the other side had views to a brick wall.[37] Researchers need to cautiously assess whether hidden factors may be occurring to the detriment of randomization. For instance, patients with higher acuity may be assigned to rooms closer to nurse stations so that they can be supervised more closely. Risks like this highlight the necessity of involving a human-in-the-loop. Humans can discover and address these outside considerations with data exploration and validation.

In the broader context of studying the value of design, Turan et al. use a **hedonic pricing model regression** to estimate the value of daylight. Spatial daylight autonomy is provided as an independent variable along with other relevant inputs such as building class, lease duration, and building age, and are considered relative to the output variable of net effective rent.[36] This illustrates the importance of controlling for outside factors; daylight plays a role, but to see it, we first need to peel back the influence of other influential variables. This is especially the case in healthcare settings, where factors like a patient's pre-existing conditions are likely to have a much greater influence on mortality rates than the architecture.

Neural networks can also provide insight into the extent to which an architectural characteristic influences health outcomes, though with limited interpretability. By conducting **ablation** and **inclusion** analyses, the relative importance of each input feature can be assessed. This method is the subject of Chapter 5.

3.3.2 Influence mapping

Other methods have been used successfully in clinical settings because they offer some degree of interpretability and reasoning about causality. **Decision trees** can be constructed using automated processes and can be used to reason about potential treatment options for patients. Decision trees can reveal causal dependencies and are presented in graphical forms that make them easy for humans to interact and reason with.[22]

Bayes nets also enable causal reasoning and have been used widely in healthcare settings. Arora et al. find that this is because they make it easy to visualize relationships between variables and because they translate easily into deployable decision models.[3]

3.3.3 Similarity mapping

Many architectural elements are interrelated; larger rooms may have larger windows, which may provide more daylight, for instance. Trade-offs are equally frequent; larger rooms will lead to longer travel distances between rooms. It may be useful to used unsupervised learning techniques like **k-means clustering** to group together similar rooms based on their holistic qualities, potentially also adding additional power to regression analyses.

3.3.4 Discussion

Several data science and machine learning methods will be at our disposal if we can generate a wide dataset of healthcare architecture. However, cultural and operational nuances could unwittingly corrupt natural experiments. Omitted variables could create bias in regressions. We should push the limits of the analyses described in this section, but we should do so with the support of a human-in-the-loop to be on the lookout for these potential pitfalls.

3.4 Decision-making

Ultimately, the results of these analyses need to make it back to the designer if they are to influence the design of new buildings. In this section, I present several methods for encouraging this feedback loop. It is an oversimplification to present these methods along a continuum, but I do so here for clarity. At one end, tools for data discovery and validation rely on computation but are driven by human operators. On the other end of the spectrum, optioneering design spaces may be defined by a human but be explored by machines. In the middle, there is the potential for design heuristics or human-machine question asking.

3.4.1 Data discovery and validation

New architectural datasets will enable new kinds of data visualization. Researchers can visualize health outcomes as overlays to 3D models of the hospital, allowing them to identify trends and patterns not visible in other forms. I provide additional detail on this topic in the form of a case study in Chapter 4.

3.4.2 Design heuristics

As the statistical methods that are described in the previous section are deployed across deeper and wider datasets, there is the potential to codify the findings into best practices. These could come in the form of hard requirements like building codes, or feed into design-criteria similar to how studies are used today. Fu et al. identify design "principles, guidelines, and heuristics" as three terms that are often used to "codify and formalize design knowledge so that innovative, archival practices may be communicated and used to advance design science and solve future design problems".[12]

3.4.3 Optioneering

To a degree, these heuristics can be translated into fitness metrics for generative design processes, enabling a process of optioneering. This does not obviate the need for human involvement; it is still necessary to frame design problems and goals, define design criteria, and maintain a watchful eye for heuristics that are misapplied. In an optioneering process, we run the risk of portraying more confidence or generalizability than the statistical analyses actually provide.

While computers can iterate through millions of design options and score each against established design criteria, they tend to stumble when faced with edge cases and disappoint when it comes to creative capacity. An uneven fitness landscape and goals that are often mutually exclusive make matters more complicated; a hospital CFO may want to minimize construction cost while a doctor may advocate for larger patient rooms to improve patient experience. The design process can be more about politics than about optimization. Computers need humans to exercise creativity, set thorough constraints, and to interpret their output.

Optioneering is difficult because of the vast potential sources of design criteria that an architect must consider. These criteria come from building codes, programming documents, letters of intent, community meetings, conversations between clients and staff, focus groups, studies, simulations, geospatial analyses, and precedents, to name a few. Many of the criteria that are crucial to improving health outcomes come in the form of journal articles or best practice compendiums. Design processes rarely leave enough time for architects to read and take advantage of these sources. Computers could help by rapidly surveying these sources to identify relevant information, but would need a framework to understand stories and convert them into internal representations to do so.

3.4.4 Recipe following and question asking

Winston and Holmes offer one such framework in their paper, *The Genesis Enterprise*.[39] Genesis is a program that takes short stories and translates them into a robust internal representation. Yang and Winston illustrate how Genesis enables computational recipe following and question asking.[40] In particular, they show how a computer can be presented with a task, follow recipes for certain behaviors, and ask another expert for help when it gets stuck. The architectural goals listed above could be interpreted via story understanding and called via recipe following.

Performance criteria, constraints, and conceptual strategies are specified. A computer can generate many of these constraints automatically, for instance, by consolidating relevant studies or running new regressions on data that is relevant to the problem at hand. A human can establish other goals, such as facilitating meaningful conversations between doctors and patients.

These goals could be integrated with a generative design process, in which many design options are generated with the goal of sampling a large design space. If the design space is well constrained, a computer can iterate through options much quicker than a human. However, constraints are often inadequate or too restrictive, and humans may be able to intervene to make adjustments to these constraints. Each design option can be evaluated based on the design criteria by both humans and machines.

3.5 Discussion

There is no silver bullet approach to optimizing design. Architecture has multiple stakeholders, and arriving at a single design solution requires negotiation. Today, these negotiations happen within an ecosystem of uncertainty, and there is more that we can and should do to build up a more robust evidence base. But even once we do so, we need to be aware that these studies will never provide us with a complete representation of the world and how it functions. We should use generative design engines to explore design spaces because they can help us track performance across a multitude of factors that we can't track on our own, but we should be on the lookout for areas where our design constraints are too strict or uncertain. We should learn what drives health outcomes in our buildings and optimize for those considerations, but we should be careful not to forget elements of design that can't be quantified, whose value is not easily articulated. But by defining the value of design more broadly to include health outcomes, we can bring more considerations into the fold and improve the built environment along the way.

4. Data Discovery in Architectural Space: A 3D Frontend for Kyrix

If we want to leverage large datasets to understand phenomena that occur spatially, then we need data visualization tools for conducting data discovery and verification in three dimensions. In this chapter, I describe several steps toward this goal. First, I present a 3D frontend for the Kyrix details-on-demand system that enables developers to create 3D visualizations and interactions that take advantage of Kyrix's backend performance optimizations. Next, I describe a process for generating 3D models by extracting structured geometric and identifying data from unstructured architectural drawings. Finally, I describe how the frontend can be used for data discovery and exploration by describing visualizations that help users explore potential transmission paths of the infectious disease *c. difficle* at Massachusetts General Hospital.

4.1 Background

This research was conducted with the Data Systems Group at MIT's Computer Science and Artificial Intelligence Lab (CSAIL) in an effort to discover potential transmission paths of the infection *c. difficile* (*c. diff*). Hospital-acquired infections like *c. diff* can spread through facilities and infect patients who had come to the hospital for other injuries or illnesses, but the mechanisms and transmission paths by which they spread is unknown. CSAIL's efforts aim to shed light on these transmission paths by enabling infectious disease experts to navigate large amounts of patient data. The 3D frontend described in this chapter layers in spatial information so that users can visualize the spread within its spatial context.

4.1.1 Spatial epidemiology

This investigation is in the spirit of John Snow's mapping of the Broad Street cholera outbreak of 1854, which visualized health data spatially on a map and ultimately led to the discovery of cholera's previously misunderstood transmission paths.[28] This kind of spatial data visualization remains a powerful tool for studying disease transmission vectors that are not fully understood today, and we have new tools at our disposal to add in additional sophistication. Electronic medical records (EMRs) have become ubiquitous throughout hospitals in the past decade. They offer rich and voluminous representations of the world that researchers can mine for the kinds of insights that Snow discovered in 1854.

4.1.2 Challenges for architectural epidemiology

However, several challenges remain. Intuitive data visualization requires fast response times to enable fluid interaction, but EMR datasets are massive and can slow performance. Second, while advances in geospatial datasets make it easier for researchers to leverage urban data such as streets and building footprints, details about the interiors of buildings remain stored primarily in unstructured architectural drawings. This presents a barrier to tracking infections like *c. diff*, which can spread throughout the interiors of hospitals.

Architectural data is not easily processed or accessed. Information such as room size, shape, layout, and position are often stored in DWG file formats. These files consist of geometric information that a user has input through a drawing program such as Autodesk's AutoCAD. While BIM formats enable associations between geometry, spatial relationships, and room details, records of many existing buildings, including those at MGH, are stored without these embedded attributes.

This chapter proposes a method for overcoming these obstacles by 1) presenting a new 3D frontend for the Kyrix details-on-demand system that takes advantage of Kyrix's backend performance optimizations to allow for data exploration with minimal response times while accounting for the unique considerations of 3D data exploration, and 2) generating structured 3D models from unstructured CAD drawings to enable data exploration in architectural space.

4.2 3D frontend for Kyrix

The Kyrix system provides an end-to-end, general-purpose system for optimizing details-on-demand data visualizations, minimizing the burden on developers. Kyrix provides developers with a "concise yet expressive declarative language for specifying visualizations," enabling the developer to focus on designing the desired interactions while the Kyrix compiler and backend handle precomputation. This structure provides quick response times even when working with massive datasets.[33]

Kyrix currently supports pan/zoom interactions with two-dimensional interfaces, but its declarative language had previously not allowed users to create three-dimensional visualizations and interactions.[34] In this chapter, I propose a new frontend for Kyrix that enables developers to specify three-dimensional visualizations and interactions with a declarative language that mirrors that of the current frontend.

4.2.1 Kyrix 2D declarative model

Kyrix's 2D frontend uses several abstractions that the 3D frontend builds upon. Kyrix's 2D frontend uses **canvases** as the context for the visualization's geometry, **layers** to specify various types of visual encodings, **data transforms** to access data via SQL queries, **rendering functions** to map data to visual objects, **placement functions** to support faster backend fetching, and **jumps** to move between different views.[34]

To visualize architectural data using the 2D frontend, a developer could create a canvas containing an SVG floorplan. She could then add additional information to the plan, such as circles that are color-coded to indicate the number of infections in any given room. By adding jumps to each of these circles, she could allow users to access new canvases upon clicking. Jumping to this new canvas would replace the view of the floorplan with a view of another type of data visualization–a timeline view of nurse activity, for instance.

While 2D views are practical for many data types, they have significant downsides when applied to the task of navigating activities that take place in three-dimensional space. Most significantly, they do not allow users to view activities that take place over multiple floors in an intuitive way. While it is possible to implement jumps that allow users to navigate from one floor to another, this kind of transition could be disorienting for the user. Additionally, it misses the opportunity to highlight spatial relationships that occur over multiple floors.

4.2.2 Kyrix 3D declarative model

3D visualizations can use much of the same declarative language. However, some alterations are necessary to implement 3D scenes and ensure usability. In particular, visual-spatial references are useful when navigating 3D scenes. For instance, when visualizing a specific room in a hospital, it may be helpful to visually key the room into its broader context: a floor, unit, or building. The 3D frontend is designed with this consideration in mind: it assumes that zooming and jumps will occur within a persistent global scene. Jumps in Kyrix 2D allow users to navigate between canvases. However, jumps in 3D Kyrix typically allow users to view different layers within the same canvas.

A typical workflow in 3D Kyrix consists of defining a **scene** to which geometry can be added. A developer can add different types of geometry to the scene using **layers**. Layers use **transform functions** to query a database and select *which* geometry that should be added to the scene and **rendering functions** that prescribe *how* the geometry is added to the scene. For instance, a developer could create a layer consisting of only room geometries on the second level of a building, and specify a rendering function that displays these objects as white, opaque rooms within the scene. A developer may wish to present multiple layers at a time; **canvases** allow users to specify which layers are presented in the scene. **Jumps** can be added to any layer and specify which canvas the frontend will present if a user clicks on an object.

Scenes

Scenes are a new abstraction in Kyrix 3D that create a persistent environment for navigating 3D geometry between jumps. In the current implementation, scenes

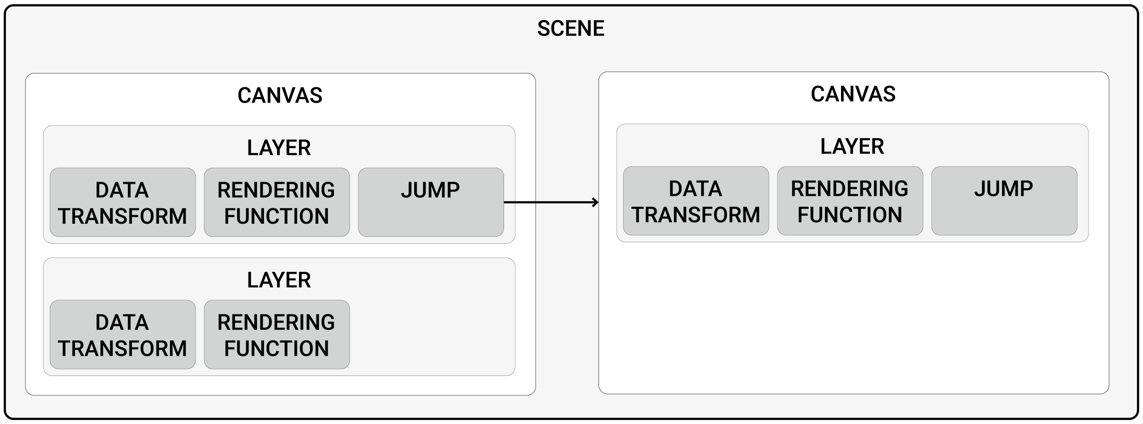

Figure 4-1: Kyrix 3D's declarative language mirrors that of Kyrix 2D, but adds a scene abstraction to enable a persistent environment when the user jumps between canvases.

are specified using the three.js 3D library.[1] Developers can add camera controls to a scene to define how a user zooms, pans, and navigates. Developers can also control the scene's visual appearance by adding elements like lighting and fog.

Canvases

In 3D Kyrix, canvases are used to declare which layers are visible in a scene. A typical canvas specification contains a list of layers to be rendered, along with any 2D user interface elements that should be presented, such as a title or subtitle. Unlike in 2D Kyrix, the scene persists when new canvases are called. This enables the user to stay oriented relative to the rest of the building as details are added or removed from the scene.

Layers

Each layer defines a set of geometric objects that should be added to a scene, along with specifications that define how the geometries should be visualized and how users can interact with those objects.

In a typical implementation of an architectural visualization, there could be:

1) a layer for rooms to allow users to interact with data associated with each room

2) a layer for building envelopes to enable users to interact with aggregated data for each building or to provide visual context for the room layer

3) a layer for static contextual information like a ground plane or site.

The developer specifies which geometries should be added to the scene by defining a data transform function for each layer. The developer specifies the appearance of objects on each layer with a rendering function. For instance, a developer could use a transform function to select only rooms that a certain patient has visited, and could then use a rendering function to color code those rooms based on the number of infections present in each room. A developer can also add a jump to the layer, which specifies which canvas loads when a user clicks on any object in the scene.

Data transforms

Data transforms define which data is retrieved from the backend for any given layer. A developer can specify that data should only be presented from a certain building, floor number, or geometry type. The developer can also provide a predicate that filters the data according to alternative conditions. Just like in Kyrix 2d, data transforms consist of SQL queries to fetch raw data.

Rendering functions

Rendering functions control the appearance of geometric objects on each layer and define how they are added to the scene. The rendering function also controls the height of objects and whether or not users can interact with them. For instance, if the primary focus of a visualization is patient rooms, then the layer containing patient rooms could have a rendering function that displays the objects as opaque and white. An additional layer for building envelopes could also be included in the canvas, and its rendering function could specify that the objects have a lower opacity and should not interact with the mouse.

A user may specify a color or color function for any layer. For instance, color may be applied along a gradient to visualize the number of infections present in each room.

Placement functions

Placement functions are not used in the current implementation. Instead, the back-end fetches data according to the transform function specified in a given layer.

Jumps

Jumps can be added to a layer and specify the canvas to view when an object is clicked, along with any associated transitions.

Discussion

Kyrix 3D's declarative language mirrors that of Kyrix's original frontend while accounting for considerations that are unique to navigating data in architectural space. The current implementation tests the flexibility of the frontend in architectural and campus settings. Still, it has not yet been tested on urban settings where larger numbers of geometric objects could cause performance issues. Implementing placement functions that take camera perspective, orbiting, and panning functionality into account provides one potential avenue to extend the frontend for this functionality.

4.3 Building an associative 3D model from unstructured CAD plans

The declarative language described in the previous section requires users to provide geometric data that is structured in such a way that it can be used to construct a three-dimensional model, which is a nonstandard format for architectural drawings. To match EMR data with architectural data, a method is needed to associate identifying information with each geometry in the 3D model, such as room number, floor number, and building name. This section describes a process of building a 3D model by extracting geometric data and its corresponding identification information from CAD plans in which no structured association exists.

First, the relevant room outline geometry is manually identified in the CAD plan, cleaned, and converted into a JSON object. Next, an attempt is made to associate room names, floor numbers, and building names with each room geometry. The geometry and the associated data are output in table form, which can then be used to reconstruct a 3D model using three.js.

Extracting formatted geometry from CAD drawings

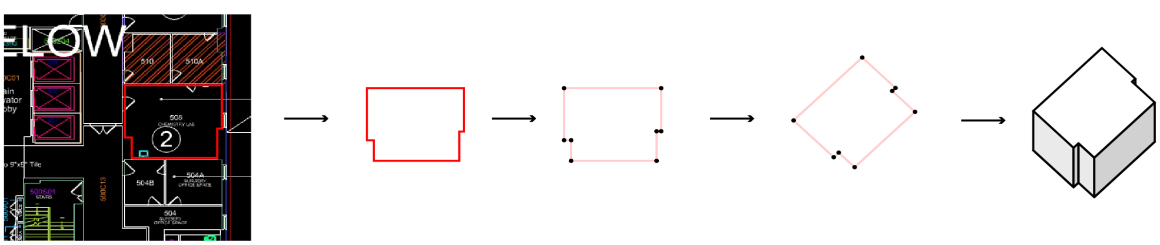

Figure 4-2: Outlines of rooms are extracted from the CAD drawings, encoded as JSON objects, and extruded into three dimensional geometries in the front-end.

First, it is necessary to extract geometry from the CAD drawings that can be used to generate the 3D model in the frontend. In this implementation, closed polylines were extracted from the CAD plans that could then be extruded in the frontend to generate 3D volumes.

There are several obstacles to automating this process. First, CAD drawings contain many types of information that are not relevant to the task at hand; geometries and annotations such as walls, lighting, furniture, fixtures, and labels need to be ignored.[2] Second, no explicit representation of each room's outline is guaranteed to exist in the drawing, making it difficult to automate extraction of these geometries. Rooms may be implied by individual lines that make up the faces of walls, but these lines may have no explicit relationship to one another in the drawing file. Gaps for windows and doors may further complicate the process of automating room outline detection. Several studies demonstrate advances in automating extraction of room boundaries from floor plans in specific conditions, but it is a problem that has not been solved universally.[32]

MGH's CAD plans contained polyline outlines of most rooms. CAD drawings are often organized with a layer table, into which a human drawer sorts certain types of geometries. This table can later be used to filter out irrelevant geometry. For instance, annotations may be kept on an annotation layer, while furniture may be kept on a furniture layer. MGH's drawings included a layer that contained outlines of each room and building, making it straightforward to isolate these geometries by simply selecting by that layer.

Some rooms did not have associated room outlines, and these needed to be identified and drawn by a human technician. Additional information was also present on the layer and needed to be filtered out, such as points, lines, and text. These could be selected and deleted using native selection features in Rhinoceros 3D. Polylines that were under a threshold square footage were also removed from the selection to ignore closets, plumbing stacks, and similar spaces that were not of interest. The result of this process was a cleaned list of polyline objects corresponding to each room in the floorplan.

Associating room names with room outlines

In these drawings, room outline geometry is not explicitly associated with room identities. Instead, room names and numbers are labeled as text objects and are often located within the room outlines. To determine which room label was associated with each room, a Grasshopper script was written to determine whether or not a text label was located within a given outline. If one text label was located within an outline, the value of that label's text was associated with the room outline. In cases where room labels were too small to be located inside the room outline, the drafter may have located the room label outside of the room and used a leader line to indicate the room it was associated with, causing this method to fail. In cases where more or fewer than one label was associated with each room, the user was notified so that they could manually adjust the labeling.

Associating floor levels with room outlines

Each geometry in the 3D model needed to be associated with the floor number that the geometry was located on. The CAD files were organized so that each CAD file contained information from a single floor. Each geometry was associated with a specific floor level in accordance with the file in which it was located.

Associating building names with room outlines

In order to associate a building name with each geometry, it is necessary to create closed polyline outlines of each building. While the CAD drawings sometimes contained this information on an associated layer, significant manual drafting was required to generate these outlines. Each building outline was stored on a unique layer named to match the building. These outlines were added to each floor plan file and varied from file to file only where the building envelope also varied. Each room's center point was tested for containment in each building's outline, and associated with any outline in which it was contained.

Data export

Each geometry was exported in a JSON format, and included the following information: 1) a **room name** stored as a string, 2) a **room level** stored as a number, and 3) a **building name** stored as a string. This data was exported to CSV using native export functionality in Grasshopper.

Creating a 3D model from room outlines

A rendering function used by each layer in the Kyrix 3D frontend adds a geometry to the scene by 1) parsing the JSON list of points, 2) generating a three.js polyline, 3) vertically extruding the polyline to create a 3D volume based on a height specified in the rendering function. Because points in the CAD plans were all had heights of 0, these points were translated vertically as a function of the level that the plan was on and a user-defined floor height.

4.3.1 Challenges and next steps

The process described above performed well on the given set of CAD drawings, but may not extend well to other drawing sources without adaptation. For instance, all of the plans used in this case study were in a consistent format with consistent origins and layer structures, making some portions of the cleaning process automatable. This may not always be the case. Taking advantage of recent developments in automatic scene digitization provides one potential avenue for overcoming this barrier. As new buildings are designed with BIM software such as Autodesk's REVIT, the necessity for this scene recognition will be obviated, and instead, simpler but separate methods to extract relevant geometric data from BIM models will be necessary.

4.4 Visualizations of *c. difficle* events at MGH

In this section, I describe how the methods presented in this chapter were deployed with data from MGH to support research into transmission vectors of *c. diff*. A series of visualizations were developed using the Kyrix 3D frontend to enable users to explore the campus as a whole and surface macro-level trends, to hone in on specific levels and units to understand trends within individual rooms, and finally to view an individual or collection of individuals' movement across the campus. The resulting visualizations make use of many aspects of Kyrix 3D's declarative language.

4.4.1 Visualizing all buildings

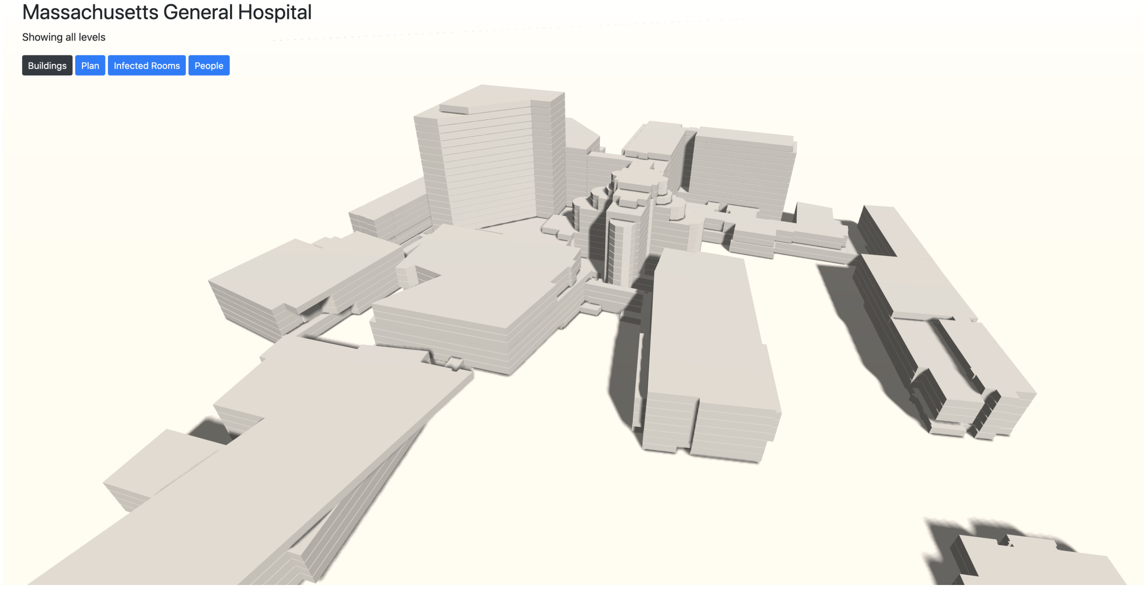

Figure 4-3: Kyrix 3D visualization showing all buildings on MGH's campus.

When a user begins to navigate the data, the frontend presents an initial view that offers a high-level over of the MGH campus. This visualization provides an opportunity for the user to orient themselves spatially on the campus. A user may come to the dashboard to investigate events in a predetermined unit of the hospital, or they may wish to engage in a more exploratory analysis to understand trends or anomalies across the campus as a whole. This view accounts for both scenarios, presenting the user with a choice to navigate quickly to a specific unit of interest, or to select a metric to visualize across the campus as a whole.

The view is constructed as a canvas with a single layer containing geometry for each level of the building. The rendering function visualizes these objects as opaque and enables interaction; on hover, these objects provide identifying information such as the building name, level, and number of infections present over a pre-specified period. Upon clicking any of these geometries, the user triggers a jump to a canvas that provides room level information for the selected floor.

Alternatively, the user may wish to color-code each level object based on a metric such as the number of infections that occurred on that level. UI elements such as buttons allow users to jump to a slight variation of this canvas that applies a rendering function that color codes the level objects based on a gradient.

4.4.2 Visualizing patient data by room

It is possible that *c. diff* spreads in specific rooms or on specific surfaces, and aggregating the results to units may not provide high enough resolution to observe these patterns, which may occur only in a single room of a level. For this reason, it is useful to be able to identify the specific rooms that a patient or staff has entered. To accommodate this scale of investigation, a view is provided that allows users to visualize each room with color coding according to individual metrics across an entire floor. There are more than 20,000 rooms at MGH, making it difficult and

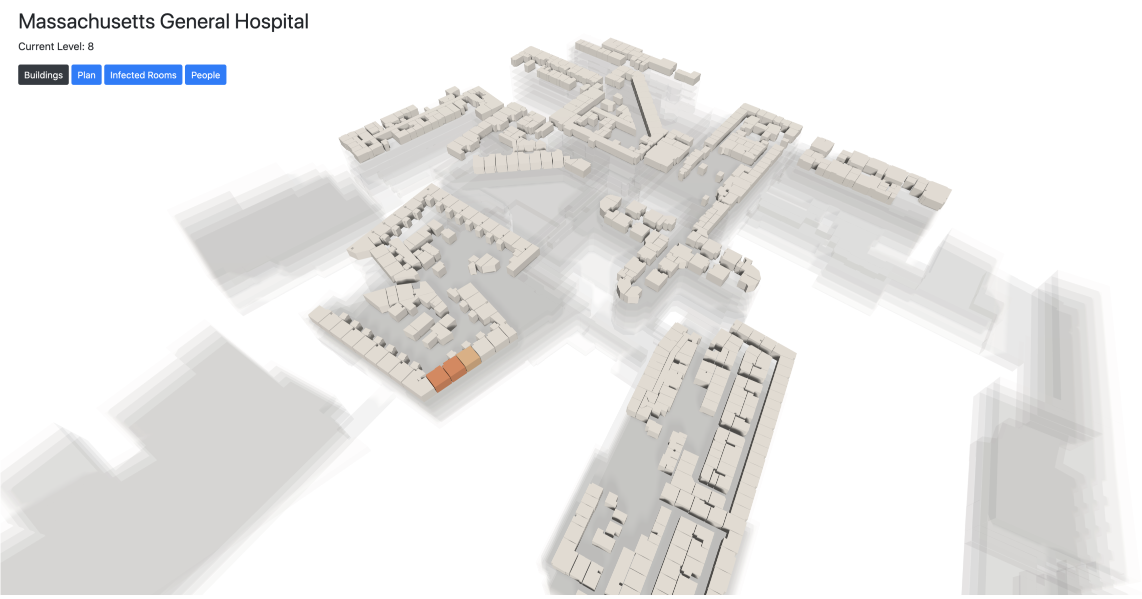

Figure 4-4: Users can visualize patient data for each room by viewing each level individually.

overwhelming to view all of them at the same time. Providing rooms for only one level at a time improves legibility.

The view is constructed as a canvas with two layers: one for room objects, and one for level objects. The room objects are the primary subject of this visualization and are color-coded with a rendering function indicating the number of infections that occurred in each room. These room objects are clickable and trigger a jump to a visualization that allows users to assess which other rooms patients and staff who visited this room also visited.

The second layer serves primarily to provide context for the visualization and consists of level objects. The transform function generates an SQL query that returns only objects that are below the currently selected level. The rendering function for this layer specifies that the objects have a low opacity so that they visually recede. It also prevents them from being clickable to avoid any interference, and prevents them from casting shadows to avoid visual noise.

4.4.3 Visualizing accumulated staff and patient activity over multiple floors

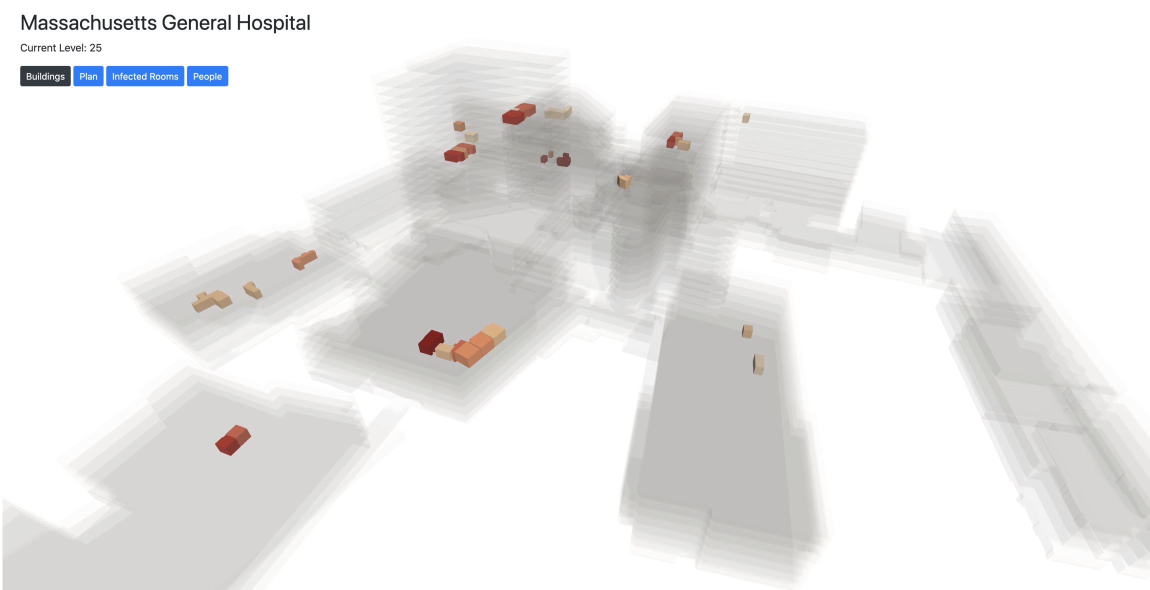

Figure 4-5: Users can visualize each room across the campus that a subset of individuals have visited.

Infected individuals, who may be either patients or staff, are not necessarily constrained to moving around a single level. Patients may travel to centralized resources such as x-ray rooms, labs, or consultation rooms. Activities like medication dispensing, consultations, and testing may be encoded in EMRs as point events recorded with timestamps and locations. Similarly, staff may have meetings or take breaks in different buildings or on different floors. Each of these movements presents a potential transmission vector, and it could be useful to view this accumulated travel without being restrained to viewing a single floor at a time or by the low resolution of only viewing individual levels. For this reason, a view that allows users to view accumulated staff and patient activity in individual rooms over multiple floors provides a useful means of studying these movements.

Similar to the previous visualization, this view is constructed as a canvas with two layers: one for room objects, and a second for level objects. The room objects are the primary focus, and a transform function is used to select only those rooms that have been visited by the subset of patients and staff specified prior to the jump. They are color-coded as specified using a rendering function. Level objects are present only to orient the viewer and are handled with the same rendering function as described in the previous view, with the exception that all levels are presented to provide the full outline of the building envelope for context.

4.5 Conclusions

This case study demonstrates that the Kyrix 3D frontend is flexible enough to accommodate several types of data visualizations and their associated tasks: high-level data discovery at the scale of a campus, detailed exploration limited by geometric constraints such as floor level, and views that highlight selections based on metric filtering criteria. These interactions cater to humans' abilities to recognize patterns, validate data, frame questions, and identify omitted variables.

4.5.1 Contributions

Through this analysis, I 1) Implemented a 3D frontend for Kyrix, enabling users to create interactive 3D visualizations using a flexible, declarative language, 2) illustrated functionality through a case study at MGH, and 3) framed the problem of integrating CAD drawings with electronic medical record data.

4.5.2 Next steps

This investigation is limited by the type of data collected; activity times and locations are recorded in the EMR only when specific events occurred, and not continuously, as is the case with RTLS data. This means that analysis of transmissions that occur between these events, such as in a hallway or elevator, are difficult to track.

Additional geometric data could be extracted from the floorplans to build a more robust and flexible 3D model. For instance, if hallways were encoded as pathways, then potential circulation patterns could be presented and used to approximate the kind of information that would otherwise come from RTLS data.

Over time and as these visualizations are used by humans to identify patterns and anomalies, these visualizations could also be coded to learn and search for the same kinds of trends that humans pick up on. In this sense, interactive visualizations could serve as a tool for humans to leverage machine intelligence and also for machines to leverage human intelligence.

5. Case Study: Neural Network Ablation Analysis

We experience architecture on multiple sensory, spatial, and temporal levels; the unique experiences that we can have in space are limitless, and so too are the ways that we can analyze and encode these spaces' characteristics. If we hope to be able to find the signal in the noise, then we need methods for considering multiple quantitative spatial characterizations at a time and surfacing those that are most relevant to predicting health outcomes. By combining spatial analytics with clinical machine learning methods, we can work toward identifying spatial characteristics that are linked to health outcomes and potentially predict the performance of different configurations before construction.

In this case study, I take several steps toward the goal of building a framework that enables clinicians and architects to make evidence-based decisions about their built environments. The process for completing this analysis consists of 1) generating a synthetic data set of architectural and health outcome data 2) encoding architectural characteristics as numeric features, 3) constructing a fully-connected neural network with spatial characteristics as inputs and health outcomes as outputs, and 4) performing an ablation analysis to determine which, if any, features most contributed to predicting health outcomes in the model.

5.1 Synthetic data set

Neural networks require large datasets to learn from and predict; no such dataset yet exists for healthcare architecture. For the purposes of this analysis, I generated synthetic data to demonstrate both the feasibility of building structured datasets of qualitative architectural information and how these datasets could be used in a neural network. The results of this analysis, therefore, do not provide insight into relationships in the real world. Instead, the synthetic data enables us to prototype models to find and address challenges before actual data is available.

5.1.1 Unit of analysis: patient room

To maximize the number of samples and variation in the data, I selected the patient room as the unit of analysis. Larger units such as a building, level, or operational unit (i.e., intensive care unit, emergency room) dilute variation that could otherwise be observed. For instance, rooms at the end of a hallway may be more private than rooms with more traffic outside of them, a relationship that would be lost if analyzed at the scale of the building. Smaller units of analysis, such as a grid of individual square feet, are challenging to associate with individual patients and their outcomes and are therefore too high resolution. To that end, the synthetic dataset consisted of observations for each patient room.

5.1.2 Generative design engine

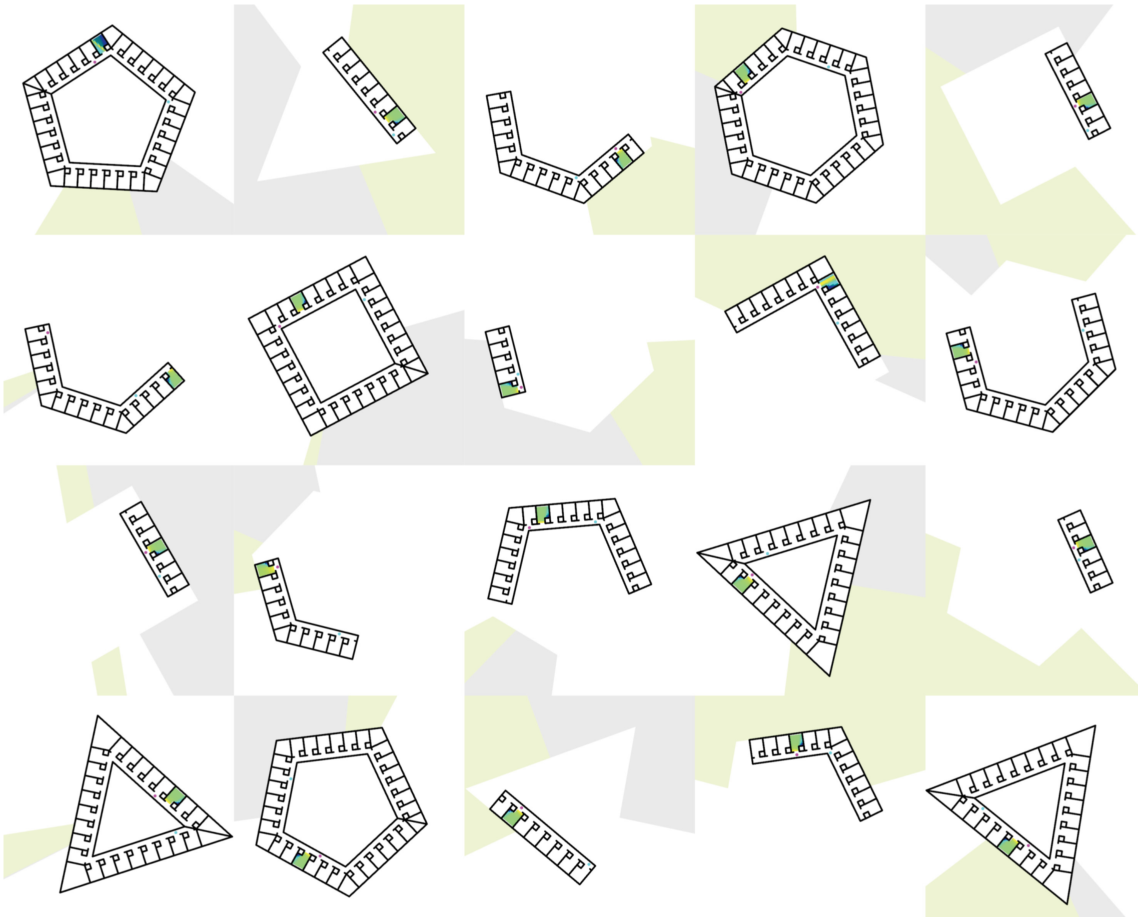

Figure 5-1: Synthetic floor plans generated through a generative design model illustrating variation in size, shape, topology, view, and room locations.

I utilized Rhinoceros and Grasshopper to develop a generative model for hospital floor plans. The model enabled parameters such as number of rooms, circulation topology, exterior views, nurse station location, elevator location, and orientation to be combined to generate over 5,000 unique plans.

5.1.3 Automated spatial analysis

The generative model was paired with an analysis engine in grasshopper, which recorded the results of spatial analyses for each room in each floor plan. These analyses included 1) travel distances to the nearest elevator and nurse station, 2) isovist area calculations at the patient bed and patient room door, 3) the view outside each window, 4) room depth, and 5) room area.

5.1.4 Synthetic health outcomes

I generated synthetic health outcome data that mirrored the types of patient data typically collected by hospitals and analyzed in related literature. These metrics consisted of 1) complication rates, 2) medical errors, 3) pain medicine intake, and 4) length of stay.

Health outcome metrics

Complication rates correspond to the number of avoidable adverse incidents that occur during a patients' stay in the hospital. These can include hospital-acquired infections, cardiac arrest, and unplanned admission to intensive care units. They may be influenced by spatial characteristics that affect team communication or patient supervision, such as travel distances and visibility.

Medical errors refer to avoidable errors in diagnosis or dispensing of medication. They may be influenced by spatial characteristics that affect staff concentration and fatigue, such as lighting and travel distances.

Pain medicine intake refers to the number of doses that a patient takes of pain medication per day. The number of doses is an indicator of a patient's discomfort, which may also be related to their anxiety levels. This may be influenced by spatial characteristics such as views to nature and exposure to noise (as may be the case in rooms that are close to nurse stations or elevators).

Length of stay refers to the number of days that a patient spends in the hospital. This may be influenced by spatial characteristics that affect staff's ability to provide quality care or the patients' ability to relax, such as exposure to noise or proximity to nurse stations.

Synthetic health outcome generation

Each observation (room) was assigned a value for each of these health outcome metrics based on relationships demonstrated in evidence-based design literature. For instance, views of nature and quiet environments may reduce discomfort and lead to lower pain medicine requests. Therefore, rooms that had views to nature or longer distances to noise generating zones such as elevators were assigned lower lengths of stay than those with views to hardscapes or were close to elevators. Values were assigned using the rules indicated in figure 5-2.

	Length of Stay	Pain Medicine Requests	Medical Error Rate	Complication Rate
Starting Value	12 days	10 doses / day	5% probability	10% probability
View				
Greenery	- 2 day	- 3 dose / day	- 1% probability	
Hardscape	- 1 days		- 1% probability	
Building				
Distance to Nurse Station < 25'	- 2 days			
Distance to Elevator < 50'	+ 0.5 days	+ 0.5 doses per day		+ 2% probability
Room Depth				+ 5% probability
Size < 450 sf			- 5% probability	

Figure 5-2: Synthetic health data was generated based on the rules in this table.

Of course, architecture is never the sole influence of these factors. This dataset was designed to simulate real-world challenges; events such as medical errors or complications are rare and may, therefore, be more difficult to pick up in statistical analysis. Length of stay is likely to be more a function of the medical condition a patient enters the hospital for. Medical errors are likely related to operational protocols or cultural factors such as team cohesion or a patients' medical history. For this reason, Gaussian noise was added to the data to simulate real-world variation.

5.2 Feature engineering

To be used as inputs in a neural network, qualitative architectural characteristics need to be encoded as numeric values. This section describes the process by which spatial characteristics were analyzed and encoded as input nodes to the neural network.

Room depth

Room depth, a term coined by Lionel March, corresponds to the extent to which nurses are likely to walk past a patient room. For each room, a number between zero and one was generated that corresponded to the percentage of all possible travel paths that pass that room. This value served as a single input node.

Isovist analysis

For every square foot in the patient room, the weighted average area was calculated. The area in square feet was normalized to a value between zero and one. Values were recorded for the isovist weighted area at three locations in the room: at the patient's head, at the door, and at the sink. Each location's value was fed into an input node.

Views

Each room had one of three views: to greenery, a building, or a hardscape. These values were one-hot encoded; each view input node was encoded as either a zero or one, depending on whether the room's view corresponded.

Distance

For each room, the distance to the nearest 1) elevator and 2) nurse station was recorded in linear feet. This value was normalized to a value between zero and one.

Room area

For each room, the square footage was calculated and normalized to a value between zero and one.

5.3 Neural network architecture

A neural network was constructed with 1) an input layer of ten nodes consisting of the spatial features described above, 2) two hidden ReLU activation layers with 64 nodes each, and 3) an output layer of four nodes consisting of the health outcomes described above, normalized from 0-1.

5.4 Ablation analysis

An ablation analysis was conducted with the synthetic data, in which input features were sequentially left out of the model, one at the time, to assess how leaving the

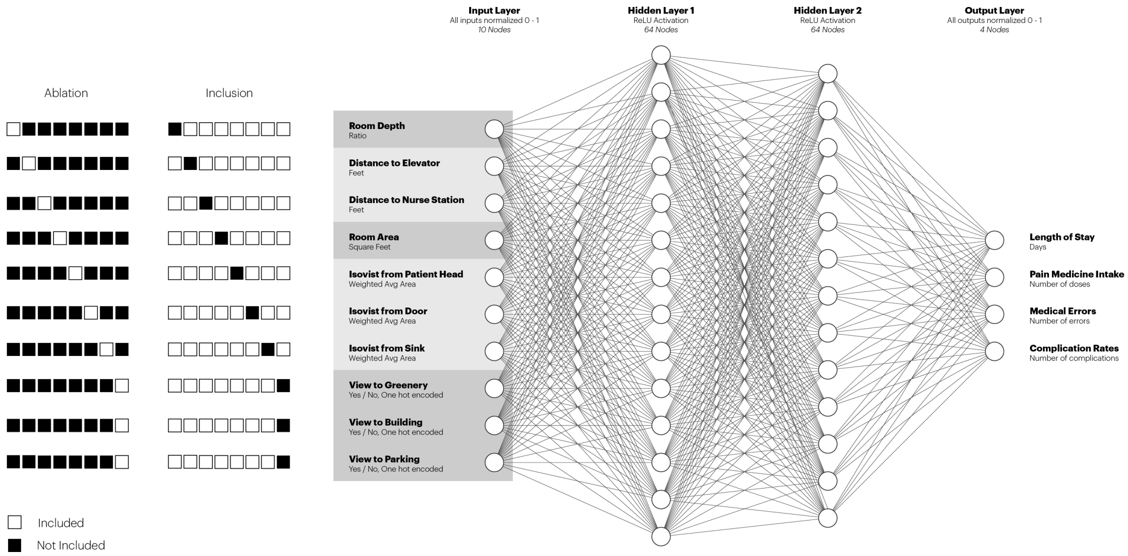

Figure 5-3: An ablation analysis was conducted using spatial features as input characteristics for a neural network with health outcomes as output features.

value out affected performance. For numeric variables, the mean square error was calculated, and for categorical variables, accuracy was calculated.

5.5 Results

roomDepth	distToElevator	distToNurseStation	roomArea	headIsovist	doorIsovist	sinkIsovist	isGreenery	isBuilding	Length of Stay MSE	Pain Medicine MSE	Medical Error Accuracy	Complications Accuracy
■									0.128	0.089	0.951	0.890
	■								0.128	**0.121**	0.949	0.890
		■							**0.214**	0.089	0.951	0.896
			■						0.128	0.089	0.951	0.890
				■					0.128	0.089	0.950	0.896
					■				0.129	0.089	0.951	0.890
						■			0.128	0.089	0.942	0.911
							■	■	**0.221**	**0.142**	0.951	0.912

Figure 5-4: Results of the ablation analysis

The results indicate that the neural network responded to some ablations, but not others. For instance, MSE for length of stay increased when *distance to nurse station* and *view types* were ablated, indicating that they contained information that helped the model perform better. However, the analysis did not see any difference when *distance to elevator* was removed from the analysis, perhaps because of the relatively small size of the influence in the synthetic data, or perhaps because this geometric relationship was inadvertently captured by another input variable.

The analysis did not appear to respond to ablation of variables that influenced *medical errors* or *complications*; the accuracies for these predictions indicate that the model consistently assumed that there were zero medical errors and zero complications. This model does not appear to be well-suited to recognizing events like these that occur only infrequently.

5.6 Conclusions

Architectural characteristics can be transformed into feature vectors that can be used as inputs to several data science analysis and prediction models, including neural networks. This case study illustrates one such approach using synthetic data and suggests that future work could prove fruitful.

5.6.1 Contributions

Through this analysis, I 1) created a synthetic dataset of architectural and health outcome by implementing a generative process, 2) implemented a feature engineering process for architectural data, illustrating how architectural characteristics can be used as inputs in data science applications, 3) implemented a neural network that predicts health outcomes as outputs from architectural characteristics as inputs, and 4) performed an ablation analysis using the synthetic dataset with the neural network.

5.6.2 Next steps

This current analysis was limited to only ten input nodes and four output nodes. In practice, it would be better to include a much wider palette of architectural characteristics: materials, daylight autonomy, isovist connectivity, room shape, orientation, adjacencies, to name a few. Inputs should also ideally include information about a patient's medical history, staff, or treatment plan.

It should be noted that neural networks are currently limited in terms of their interpretability and their ability to provide insight into causality. There is always the risk of observing and acting upon correlations that are not causal. Geometric considerations compound this risk; many architectural characteristics are geometrically intertwined. Rooms at the end of hallways are likely to be more private and also likely to be further away from nurse stations, but proximity to nurse stations is more likely to be a driver of quality patient care than is privacy. Covariances like these riddle architectural analyses, and should be considered in any investigation.

Because this analysis uses synthetic data, the results do not yet provide insight into the nature of the relationship between architecture and health. However, this proof of concept illustrates that with the right data, neural networks are worth investigating further. With access to wider and larger datasets, there is the potential to use a method like the one described here to not only learn from existing data but also to potentially predict the performance of future floorplans.

6. Conclusion

Hospitals present a unique opportunity in the discipline of architecture to demonstrate the value of design. Decades of evidence-based design research indicates that architecture affects our health, but these findings do not guarantee generalizability. If we want to build out a more robust model of *architectural epidemiology*, then we need to take advantage of opportunities that analysis at scale provides us: the ability to account for omitted variable bias, to search for natural experiments, and to learn from contexts and situations are most similar to the design task at hand. To achieve analysis at scale, we need data at scale.

Robust electronic medical records have matured; what remains is to build a large scale data of architectural characteristics that researchers can use in analyses. We need to overcome several challenges to do so: structured data must be extracted from a heterogeneous body of unstructured architectural drawings, and this data needs to be wide enough that it captures the qualitative aspects of our environments that affect our health.

Once we have these datasets, we need methods to validate, explore, and mine for insight. As we learn from buildings, these methods need to take advantage of humans' abilities to recognize factors that fall outside the realm of what current datasets capture and to define research questions. As we design buildings, these methods need to account for humans' abilities to define relevant fitness criteria and design spaces. At the same time, we need computational methods to reduce bottlenecks and enable us to deal with the challenges of big data. We need data visualizations that allow us to work with massive datasets in realtime. We need the ability to weigh a wide range of factors at once and to evaluate the performance of large numbers of design options.

These efforts have the benefit of being able to build upon established research efforts in several related fields. Evidence-based design research provides a foundation for understanding architectural characteristics that affect health, and researchers have demonstrated many methods for testing hypotheses via individual research studies. Space Syntax provides methods for quantifying qualitative aspects of the built environment and has a rich history of using these analyses to learn about how architecture affects our health and behaviors. What remains is for these disciplines to adapt to opportunities afforded by more robust datasets.

Researchers in commercial real-estate have made progress on this front. In her 2018 thesis considering the role of AI and machine learning, Jennifer Conway identified several areas of active application in practice, including in sales tools, property management, analytics, contracts, lending, and valuation.[8] These applications highlight the challenges of working with data related to the built environment and propose ways forward. These methods can and should be enriched by data resulting from spatial analysis. Definitions of value should be extended to include not only dollars and cents but also how buildings affect our health.

6.1 Contributions

The preceding work took several steps toward the goal of building upon work in evidence-based design, space syntax, and machine learning applications in real estate to define a framework of architectural epidemiology.

1) Conducted a literature review to identify criteria for a framework of *architectural epidemiology*

2) Proposed a framework of *architectural epidemiology* to learn from large health and architectural datasets

3) Implemented a 3D frontend that enables developers to validate and explore health outcome data in architectural space

4) Implemented a neural network ablation analysis with synthetic data to illustrate how architectural data can be used in data science analyses

6.2 Next Steps

The efforts described in this thesis suggest that combining structured architectural datasets with computational analysis in ways that take advantage of human intuition holds the potential to improve our ability to design buildings that will enhance our health. Still, much work remains.

Most pressingly, we need to develop large scale architectural datasets that capture a wide range of environmental characteristics. This is a prerequisite for substantive data analysis and discovery. To do so, we'll need to develop consistent, standardized ways of analyzing spatial characteristics and processing floorplans in ways that can be at least partially automated. This is a long-term project; we'll need to continue to add features as we learn more about which design aspects are important.

With this data in hand, we will be able to test a growing body of data science and machine learning techniques to identify relationships, establish heuristics, and potentially drive generative design processes. Significant work remains in establishing and testing these methods.

Critically, these insights need to feed back into the design process. We need to do so in a way that limits information overload for designers while making it easy to challenge assumptions and conclusions that derive from automated analyses. This is not a small task. It will require iteration and testing, perhaps comparing the outcomes of human-driven design processes with those of generative or computer-assisted processes.

The question of optimization will remain elusive. In order to optimize, we need to agree on what to optimize for, and in doing so, we risk optimizing for aspects of design that are quantifiable rather than those that elude analysis. It is my hope that by bringing more qualitative aspects of design into the fold during discussions about the value of design that we will be empowered to design buildings that can help us be happier and healthier.

Bibliography

[1] Three.js JavaScript 3D library, May 2020. original-date: 2010-03-23T18:58:01Z.

[2] Sheraz Ahmed, Markus Weber, Marcus Liwicki, and Andreas Dengel. Text/Graphics Segmentation in Architectural Floor Plans. In *2011 International Conference on Document Analysis and Recognition*, pages 734–738, Beijing, China, September 2011. IEEE.

[3] Paul Arora, Devon Boyne, Justin J. Slater, Alind Gupta, Darren R. Brenner, and Marek J. Druzdzel. Bayesian Networks for Risk Prediction Using Real-World Data: A Tool for Precision Medicine. *Value in Health*, 22(4):439–445, April 2019.

[4] Franklin Becker and Stephanie Douglass. The Ecology of the Patient Visit: Physical Attractiveness, Waiting Times, and Perceived Quality of Care. *The Journal of Ambulatory Care Management*, 31(2):128–141, 2008. Accession Number: 00004479-200804000-00006 ISBN: 0148-9917 Type: 10.1097/01.JAC.0000314703.34795.44.

[5] Terry L Buchanan, Kenneth N Barker, J Tyrone Gibson, Bernard C Jiang, and Robert E Pearson. Illumination and errors in dispensing. *American journal of hospital pharmacy*, 48(10):2137–2145, 1991. Publisher: Oxford University Press.

[6] Andrea Chegut, Daniel Fink, and Hunter Fields. The Wide Data Experiment.

[7] HA Cohen, E Kitai, I Levy, and D Ben-Amitai. Handwashing patterns in two dermatology clinics. *Dermatology*, 205(4):358–361, 2002. Publisher: Karger Publishers.

[8] Jennifer Conway. Artificial Intelligence and Machine Learning: Current Applications in Real Estate.

[9] Stephanie J. Crowley, Clara Lee, Christine Y. Tseng, Louis F. Fogg, and Charmane I. Eastman. Combinations of Bright Light, Scheduled Dark, Sunglasses, and Melatonin to Facilitate Circadian Entrainment to Night Shift Work. *Journal of Biological Rhythms*, 18(6):513–523, 2003. _eprint: https://doi.org/10.1177/0748730403258422.

[10] Ivor D'Souza, Wei Ma, and Cindy Notobartolo. Real-Time Location Systems for Hospital Emergency Response. *IT Professional*, 13(2):37–43, March 2011.

[11] Lindsey Fay, Hui Cai, and Kevin Real. A Systematic Literature Review of Empirical Studies on Decentralized Nursing Stations. *HERD: Health Environments Research & Design Journal*, 12(1):44–68, 2019. _eprint: https://doi.org/10.1177/1937586718805222.

[12] Katherine K. Fu, Maria C. Yang, and Kristin L. Wood. Design Principles: The Foundation of Design. In *Volume 7: 27th International Conference on Design Theory and Methodology*, page V007T06A034, Boston, Massachusetts, USA, August 2015. American Society of Mechanical Engineers.

[13] Arsalan Gharaveis, D. Kirk Hamilton, Debajyoti Pati, and Mardelle Shepley. The Impact of Visibility on Teamwork, Collaborative Communication, and Security in Emergency Departments: An Exploratory Study. *HERD: Health Environments Research & Design Journal*, 11(4):37–49, 2018. _eprint: https://doi.org/10.1177/1937586717735290.

[14] Inger Hagerman, Gundars Rasmanis, Vanja Blomkvist, Roger Ulrich, Claire Anne Eriksen, and Töres Theorell. Influence of intensive coronary care acoustics on the quality of care and physiological state of patients. *International Journal of Cardiology*, 98(2):267 – 270, 2005.

[15] Saif Haq and Yang Luo. Space Syntax in Healthcare Facilities Research: A Review. *PA P E R S*, 5(4):21.

[16] Lorissa MacAllister, Craig Zimring, and Erica Ryherd. Exploring the relationships between patient room layout and patient satisfaction. *HERD: Health Environments Research & Design Journal*, 12(1):91–107, 2019. Publisher: SAGE Publications Sage CA: Los Angeles, CA.

[17] Justin Martin. *Genius of Place: The Life of Frederick Law Olmsted*. Hachette Books, May 2011. Google-Books-ID: Xiy6E0oVQ2UC.

[18] Lynn McDonald. *Florence Nightingale and Hospital Reform: Collected Works of Florence Nightingale*. Wilfrid Laurier Univ. Press, December 2012. Google-Books-ID: xYPZAgAAQBAJ.

[19] Florence Nightingale. Example of polar area diagram by Florence Nightingale (1820–1910). Public domain. Wikimedia Commons., 1858.

[20] Frederick Law Olmsted. *The Papers of Frederick Law Olmsted: The Early Boston Years, 1882–1890*. JHU Press, 1977. Google-Books-ID: UTHSAQAAQBAJ.

[21] Michelle Ossmann, Sonit Bafna, Craig Zimring, and David Murphy. Measuring the potential for concurrent targeted surveillance and general awareness. page 16.

[22] Vili Podgorelec, Peter Kokol, Bruno Stiglic, and Ivan Rozman. Decision Trees: An Overview and Their Use in Medicine. *Journal of Medical Systems*, page 20, 2002.

[23] Xiaobo Quan, Anjali Joseph, Eileen Malone, Debajyoti Pati, and Leed Ap. Healthcare Environmental Terms and Outcome Measures: An Evidence-based Design Glossary. page 71.

[24] Jonas Rehn and Kai Schuster. Clinic Design as Placebo—Using Design to Promote Healing and Support Treatments. *Behavioral Sciences*, 7(4):77, November 2017.

[25] Paloma Gonzalez Rojas. SPACE AND MOTION: Data based rules of public space pedestrian motion. page 108.

[26] Donald A. Schon. *The reflective practitioner: How professionals think in action*, volume 5126. Basic books, 1984.

[27] Mardelle McCuskey Shepley. Predesign and Postoccupancy Analysis of Staff Behavior in a Neonatal Intensive Care Unit. *Children's Health Care*, 31(3):237–253, 2002. Publisher: Taylor & Francis _eprint: https://doi.org/10.1207/S15326888CHC3103_5.

[28] Narushige Shiode, Shino Shiode, Elodie Rod-Thatcher, Sanjay Rana, and Peter Vinten-Johansen. The mortality rates and the space-time patterns of John Snow's cholera epidemic map. *International Journal of Health Geographics*, 14(1):21, December 2015.

[29] Herbert A. Simon. The Science of Design: Creating the Artificial. *Design Issues*, 4(1/2):67–82, 1988. Publisher: The MIT Press.

[30] John Snow. Original map made by John Snow in 1854. "On the Mode of Communication of Cholera." Public Domain. Wikimedia Commons., 1854.

[31] John E. Swan, Lynne D. Richardson, and James D. Hutton. Do Appealing Hospital Rooms Increase Patient Evaluations of Physicians, Nurses, and Hospital Services? *Health Care Management Review*, 28(3):254–264, 2003. Accession Number: 00004010-200307000-00006 ISBN: 0361-6274.

[32] Rui Tang, Yuhan Wang, Darren Cosker, and Wenbin Li. Automatic structural scene digitalization. *PLOS ONE*, 12(11):e0187513, November 2017.

[33] Wenbo Tao and Xiaoyu Liu. Kyrix: Interactive Visual Data Exploration at Scale. page 6.

[34] Wenbo Tao, Xiaoyu Liu, Yedi Wang, and Leilani Battle. Kyrix: Interactive Pan/Zoom Visualizations at Scale. page 12, 2019.

[35] Sara Ann Taylor, Natasha Jaques, Ehimwenma Nosakhare, Akane Sano, and Rosalind Picard. Personalized Multitask Learning for Predicting Tomorrow's Mood, Stress, and Health. *IEEE Transactions on Affective Computing*, pages 1–1, 2017.

[36] Irmak Turan, Andrea Chegut, Daniel Fink, and Christoph Reinhart. The value of daylight in office spaces. *Building and Environment*, 168:106503, January 2020.

[37] R. Ulrich. View through a window may influence recovery from surgery. *Science*, 224(4647):420–421, April 1984.

[38] Dan Willis, William W Braham, Katsuhiko Muramoto, and Daniel A Barber. *Energy accounts: Architectural representations of energy, climate, and the future*. Routledge, 2016.

[39] Patrick Henry Winston and Dylan Holmes. The Genesis Enterprise: Taking Artificial Intelligence to another Level via a Computational Account of Human Story Understanding. page 53.

[40] Zhutian Yang and Patrick Henry Winston. Learning by asking questions and learning by aligning stories: how a story-grounded problem solver can acquire knowledge. Technical report, 2018.

[41] Craig Zimring, Megan E. Denham, Jesse T. Jacob, Douglas B. Kamerow, Nancy Lenfestey, Kendall K. Hall, Altug Kasali, David Z. Cowan, and James P. Steinberg. The Role of Facility Design in Preventing Healthcare-Associated Infection: Interventions, Conclusions, and Research Needs. *HERD: Health Environments Research & Design Journal*, 7(1_suppl):127–139, 2013. _eprint: https://doi.org/10.1177/193758671300701S09.