Data Mining, Inference, and Predictive Analytics for the Built Environment with Images, Text, and WiFi Data

by

Rachelle B. Villalon
SMArchS Computation, MIT, 2008

Submitted to the Department of Architecture
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY IN THE FIELD OF ARCHITECTURE:
DESIGN AND COMPUTATION
AT THE MASSACHUSETTS INSTITUTE OF TECHNOLOGY

FEBRUARY 2017 [June 2017]

Signature of Author: Signature redacted
Department of Architecture
October 5, 2016

Certified by: Signature redacted
Takehiko Nagakura
Associate Professor of Design and Computation
Director, Design and Computation Group
Thesis Supervisor

Accepted by: Signature redacted
Sheila Kennedy
Professor of the Practice, Architecture
Chair, Department Committee on Graduate Students

READERS

Takehiko Nagakura
Associate Professor of Design and Computation,
Director of Computation Group, Department of Architecture, MIT
Thesis Supervisor

Abel Sanchez
Research Scientist, Civil and Environmental Engineering,
Executive Director of the Geospatial Data Center, MIT

Dennis R. Shelden
Associate Professor, Architecture, Georgia Tech
Director, Digital Building Laboratory

Data Mining, Inference, and Predictive Analytics for the Built Environment with Images, Text, and WiFi Data

by

Rachelle B. Villalon
SMArchS Design and Computation, MIT, 2008

Submitted To The Department of Architecture on October 5, 2016
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in the Field of Architecture: Design and Computation

Abstract

What can campus WiFi data tell us about life at MIT? What can thousands of images tell us about the way people see and occupy buildings in real-time? What can we learn about the buildings that millions of people snap pictures of and text about over time? Crowdsourcing has triggered a dramatic shift in the traditional forms of producing content. The increasing number of people contributing to the Internet has created big data that has the potential to 1) enhance the traditional forms of spatial information that the design and engineering fields are typically accustomed to; 2) yield further insights about a place or building from discovering relationships between the datasets. In this research, I explore how the Architecture, Engineering, and Construction (AEC) industry can exploit crowdsourced and non-traditional datasets. I describe its *possible* roles for the following constituents: historian, designer/city administrator, and facilities manager - roles that engage with a building's information in the past, present, and future with different goals. As part of this research, I have developed a complete software pipeline for data mining, analyzing, and visualizing large volumes of crowdsourced unstructured content about MIT and other locations from images, campus WiFi access points, and text in batch/real-time using computer vision, machine learning, and statistical modeling techniques. The software pipeline is used for exploring meaningful statistical patterns from the processed data.

Thesis Supervisor: Takehiko Nagakura

Title: Professor of Design and Computation
Director of Computation Group, Department of Architecture, MIT

Acknowledgements

I would like to thank my PhD committee: Takehiko Nagakura, Dennis Shelden, and Abel Sanchez for their feedback along the way. Your guidance has not only been invaluable throughout the dissertation, but has been indispensable since the very beginning when I entered MIT as a Master's degree candidate and/or later as a PhD student. It is my hope to continue our conversations and expand the research into future works. I would also like to thank David Barber of MIT's Security and Emergency Management Office as part of the Facilities Department, David's feedback and encouragement to use this research to assist security staff on college campuses allowed me to see one of the possible applications of this work beyond academia. I also thank Jim Harrington, Facilities Manager of MIT Architecture Facilities for his early feedback on my research direction before a line of code was ever written down.

I thank Mark Silis, Associate Vice President of the Information Systems and Technology organization at MIT, Mark provided valuable feedback and helped connect me to the gatekeepers of MIT's live WiFi data. I also thank the Big Data at the Computer Science and Artificial Intelligence Laboratory (CSAIL) group for showing and allowing me to tinker around with their tools and databases.

I would like to thank my colleagues and friends in the Computation Group as well as my non-Computation MIT friends. Thanks for your diverse perspectives and contributing to making MIT a fun place.

Finally, I thank my family. My parents Marie Jane and Olivier Villalon, who always emphasized the importance of an education. To Karen Schoucair, your support and understanding throughout my years during the PhD has been the greatest of all.

Table of Contents

Part I. Learning from Data

Chapter 1

Introduction

Looking around books and the internet on topics about "big data," it's easy to see that the general introduction to the topic begins with the massive amounts of data generated by humans and machines. On social networking websites, a person is one of 1.19 billion users on Facebook, one of 540 million users on Google Plus, one of 300 million users on Instagram, and one of 288 million users on Twitter. As a minute passes by, Facebook users are sharing 2.5 million pieces of content, YouTube users are sharing 72 hours of new video, or over 1 million hours of video content in a day, email users send 204,000,000 messages, and Google receives over 4 million search queries or 3 billion searches conducted each day[1]. All of the user uploaded data are thousands of times more than the quantity of all books printed in the U.S. Library of Congress (there's 16 million books at the Library of Congress). This does not yet include the amount of information collected from our mobile devices, my iPhone, for example, tracks my daily sleep patterns, food intake, location, and overall activity level. On top of that, add the physical objects with IP addresses that are already connected to the web from sensors embedded in manufacturing machinery to the connected temperature and lighting controllers in the home.

[1] http://aci.info/2013/11/06/the-cost-of-content-clutter-infographic/. See infographic in Appendix A.

There is a massive volume of content, if not clutter. Despite the faddish elements of "big data," the opportunity comes from analyzing and combining the data to extract insights and value[2]. With all of this information at our disposal, how can professionals in Architecture, Engineering, Construction, Facility Management (AECFM), architectural historians, and city administrators integrate multiple sources of crowd-sourced data into their workflows? This research explores the integration of data analysis and multiple streams of crowd-sourced information inside and outside of an organization as an integral part of the design decision analysis, management of a building, or for learning about a building's historical record as seen through the eyes of the public.

This dissertation explores the use of public and private data sets for the following constituents:

Designer/City Administration:

- Discover what types of buildings and facades attract visitor attention to facilitate the planning of potential future construction projects.
- Track camera positions (foot traffic) from internet photographs over time.

Facilities Management:

- Determine where crowds form and when to anticipate crowds at MIT with predictive analytics using camera positions, natural language processing for text analysis, and WiFi access points.
- Discover how a building has changed since post production due to time and weather through the analysis of photographs and the resulting 3D point cloud data/image-based model.

Historian:

- Discover how a building has changed and/or been modified over time through internet photographs and 3D point cloud data for historical documentation.
- Discover how the public perceives a building through tweets and photographs for historical documentation.

1.1 Contributions

[2] In this research, "value" is relative to the goals of each use case persona.

In this thesis, I present a range of semi-automated algorithms and techniques for data mining, analyzing, and visualizing image and text data:

1. Discovering patterns in the image and text datasets about buildings through the three use case goals.
2. A semi-automated software pipeline for
    a. Pulling images and text datasets
    b. Data cleaning
    c. Input images and text into machine learning classification algorithms
    d. Input images into Structure from Motion (SfM) computer vision algorithms for 3D reconstruction
    e. Extracting SfM camera matrix for camera pose estimation and 3D point cloud visualization
    f. Analyzing results with statistical models
    g. Visualizing results in an external web application
3. Understanding crowds formed in and out of buildings through WiFi activity.

1.2 Road Map

Part I of the dissertation, chapters 1 through 4, encompasses an exploratory analysis of the research datasets.

Chapter 1 introduces the research contributions and dissertation roadmap.

Chapter 2 is separated into three sections that address different aspects of the technologies and/or techniques that this thesis builds upon. The first section begins with a review of projects developed by companies and research communities in Design, History, and Facilities Management that implement crowdsourcing into their workflow. The second section looks at precedents in image-based 3D modeling techniques. Lastly, the chapter includes a survey of projects that visualize patterns using large datasets and then reports a selected set of user responses to the visualization tools developed for this research.

Chapter 3 analyzes the data sets and what has been discovered in the process of data mining multiple camera poses, user uploaded tags/text about MIT's Stata Center and Simmons Hall, Boston's Copley Square, as well as MIT campus WiFi access points to produce meaningful results.

Chapter 4 is a discussion on finding meaningful patterns in a very large dataset and representing information that is visually comprehensive for actionable analysis.

Part II is a technical review of the methods and algorithms that were approached for developing the software tools necessary to undertake this research.

Chapter 5 is divided between machine learning and statistical models for computer vision and natural language text processing. I describe the foundation that this research builds upon by discussing the researchers and algorithms that led to the choice in algorithmic techniques for this project such as Structure from Motion, Bag of Visual Features, SIFT, K-Means, Support Vector Machines, Non-Linear Least Squares, Multiple Regression, ARIMA, and others.

In Chapter 6, I describe the process of data mining and the challenges encountered when dealing with rich raw data from the internet. Namely, the pre-processing and data management of images and text. Both information demand a separate approach to produce useable information for image processing, 3D reconstruction, and statistical analysis.

Chapter 7 is a discussion on data visualization, I first describe the traditional reporting tools found in AEC and related fields, then introduce the custom visualization tools developed for this thesis along with user commentary.

Chapter 8 recaps the thesis contribution and identifies future research areas.

Chapter 2

Related Work

During an earthquake in China, people turned to Twitter in 2008 to share their experiences with the world. What the U.S. Geological Survey (USGS) found intriguing was that Twitter data, or "tweets" sent by users, reported faster information than the USGS sensors on an earthquake occurrence. The majority of earthquake sensors are currently installed in the United States while many "empty space"/sensorless regions around the world still exists. To that effect, USGS now implements Twitter data as part of their earthquake detection system in many languages[3] and can now receive notifications in an earthquake impacted region. Considering how crowd-sourced information can enhance existing systems and even provide other insights, how can AEC professionals, city managers, and historians leverage crowd-sourced information to either help make better informed decisions on projects for the built environment and learn about an urban space directly from users? What is there to gain by integrating industry and publicly available information through crowd sourcing? The following section introduces a couple of examples as to how the industries mentioned above have already thought about crowdsourcing in their workflows.

2.1 Crowdsourcing in Facilities Management, Architecture, and City Management

[3]"How The USGS Uses Twitter Data To Track Earthquakes | Twitter Blogs". *Blog.twitter.com*. 2015. Web. 22 Sept. 2016. <https://blog.twitter.com/2015/usgs-twitter-data-earthquake-detection>

The examples below demonstrate organizations that gather direct input from the general public to arrive at business solutions, gather content, or services. Some industries welcome the idea of crowd participation while others appear threatened, especially in the area of providing professional services for clients. In either case, the crowdsourcing platforms described in this section allude to solving a particular problem that no one person or organization could have fulfilled on their own.

The notion of crowd participation to solve a problem is not entirely new, the open source software movement is one example where many people can modify source code and fix bugs. According to a survey by Forrester, "the average time to resolve an application problem is 6.9 days for enterprise developers and 6.7 days for software vendors. Ten percent of those problems take 10 days to solve." Contrast that with open source software, the Evans Data Corporation (EDC) reported a survey of "several hundred open-source and Linux developers...[where] the average time between discovery and solution of a serious bug, for 36 percent of open-source developers is under 8 hours."[4] The open source software model does not always align with business models, but the principle of crowd participation can provide project benefits (as well as disadvantages in management and longevity). The following section introduces the crowdsourcing model as seen in other professions.

2.1.1 Crowdsourcing in Facilities Management

At the University of California at Davis, the Facilities Management Energy Conservation Office developed an interactive website so that UC Davis students, staff, and faculty "can provide real-time crowdsourced feedback" as to whether the room temperatures are within thermal comfort levels. Visitors to their Campus Energy Education Dashboard (CEED)[5] website offers a campus map and energy use intensity for some of the buildings on campus. The submitted feedback from the community becomes part of the growing database which also tracks UC

---

[4] Asay, Matt. "Open-source vs. Proprietary Software Bugs: Which Get Squashed Fastest?" *CNET*. CNET, 27 Sept. 2007. Web. 22 Mar. 2016. http://www.cnet.com/news/open-source-vs-proprietary-software-bugs-which-get-squashed-fastest/

[5] *CEFS HOME. Eco.ucdavis.edu.* Retrieved 10 August 2016, http://eco.ucdavis.edu/

Davis' unique energy system that includes electricity, chilled water, pipes to buildings for cooling and steam. UC Davis spends $30 million per year on energy, and the largest segment is attributed to heating and cooling by 50 to 60 percent. The crowdsourced data helps UC Davis monitor energy usage on campus. Figures 2.1 and 2.2 show the dashboard interface for UC Davis' energy management system.

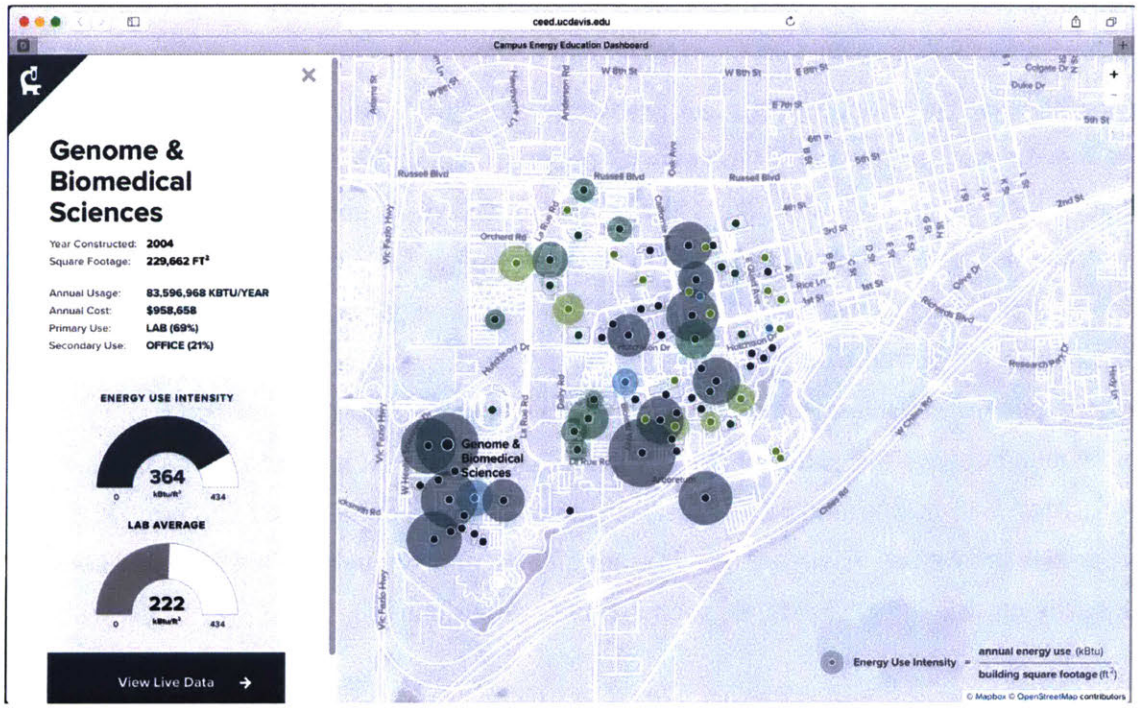


Figure 2.1 Energy usage by intensity across buildings on UC Davis' campus

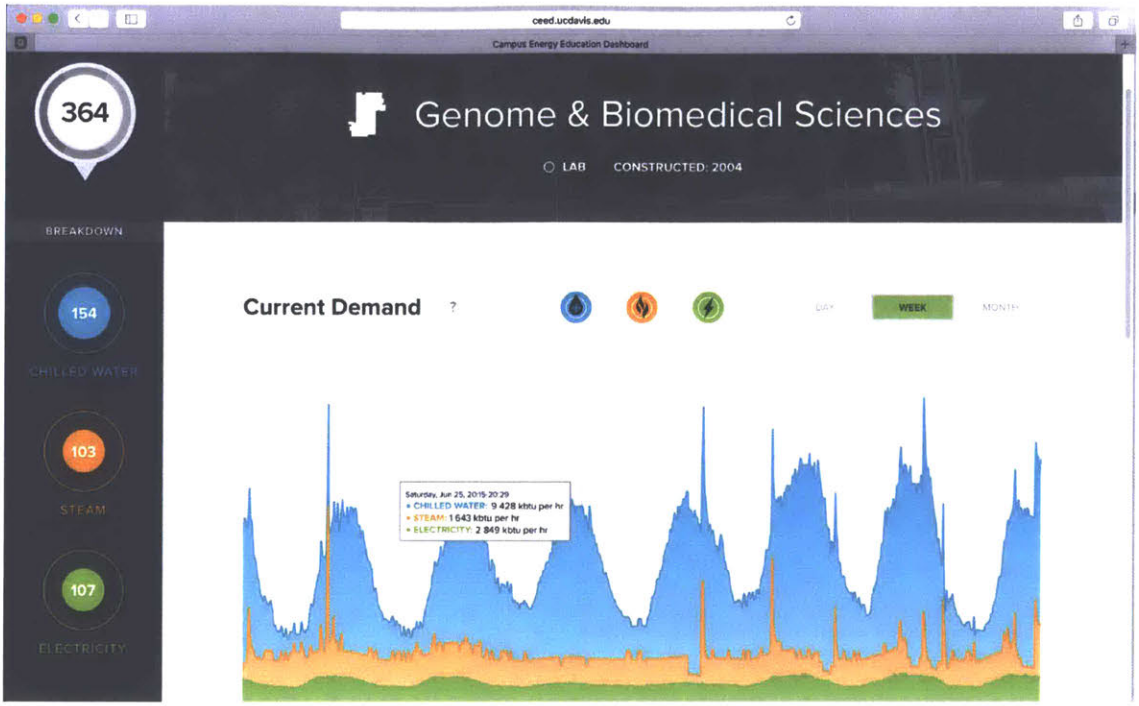


Figure 2.2 Demand graph broken down by electricity, steam, and chilled water.

2.1.2 Crowdsourcing in Architecture

Arcbazar[6] is a design competition website where clients post project proposals for architectural, interior, and landscape designers all over the world (Figure 2.3). The site allows the public to design small and medium scale projects where clients judge and offer cash rewards. The site appeals to some clients because of the multiple design submissions and the significantly lower cost to receive a design instead of hiring services from traditional design firms. Arcbazar is an example of social media changing the marketplace.

[6]As, Imdat and Takehiko Nagakura. "Architecture for the Crowd by the Crowd": Architectural Practice in the Digital Age, ArchDesign Conference, 2016, Istanbul, Turkey.

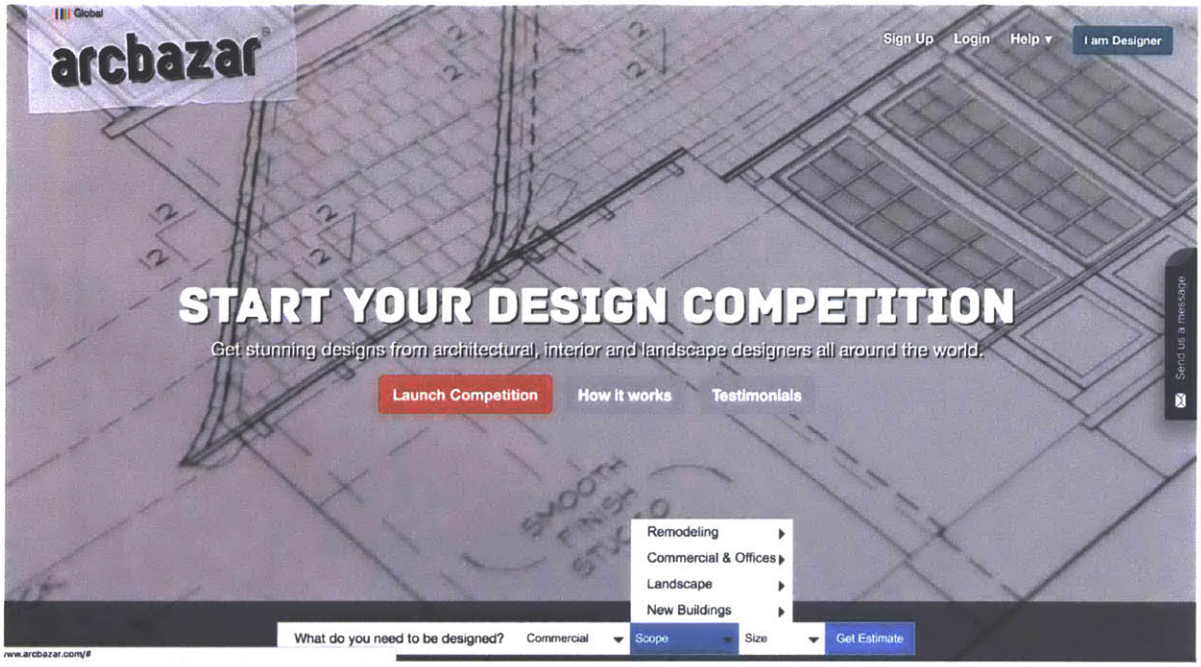


Figure 2.3 Arcbazar's design competition homepage.

2.1.3 Crowdsourcing in History

In the humanities realm, the "Living with the Railroads" project was developed to expand information across the American West about the 19th century transcontinental railroads (Figure 2.4). Catered towards railroad enthusiasts, historians, and scholars in partnership with Stanford University, the project lays the groundwork for individuals to upload and contribute data by identifying images, map, and textual content. What is usually a research project carried out by social science researchers, the participants include volunteers to share their accumulated knowledge.

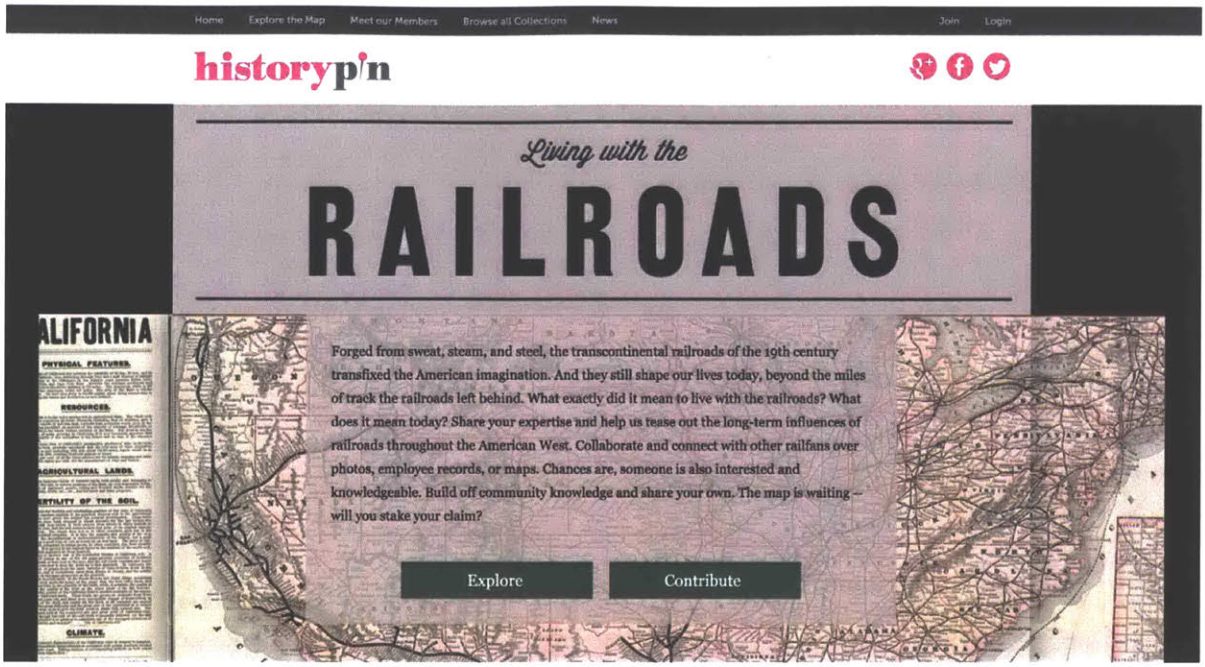


Figure 2.4 Living with the Railroads' homepage invites users to peruse the collection of railroad information or contribute knowledge.

The brief introduction to the projects mentioned above gather the public's input to either expand knowledge as in the Living with the Railroads project, create multiple options as seen in Arcbazar, or help inform decisions regarding energy usage as seen in UC Davis' efforts. What is proposed in this dissertation is a system that does not directly gather the public's feedback that inputs into a platform, but uses existing open databases to extract information deemed particularly useful for applications relevant to city management, history, and design -- importing and expanding the relevant pieces of information into existing platforms.

2.2 Precedents in Data Analysis and Visualization

Before introducing the software system for this research, exploring tools that exist in other industries can help inform how other organizations leverage data to create visualization and analysis tools demonstrating 1) the capabilities of extracting useful information from large data sets or 2) implement image based 3D modeling for capturing geometric representations in point

cloud form. The two separate features are necessary to explore particularly because the proposed software system implements data analysis and visualization with 3D point clouds.

2.2.1 Finding Patterns in Large Datasets

3D visualization can convey building components in great detail that 2D drawings cannot, yet some types of 2-dimensional information can help provide quick insights and cite patterns from very large datasets. One example of this can be found in custom business intelligence tools created by big data driven environment companies such as Netflix, where data visualization is crucial to understand the personalization of content for their customers:

> Good visualization helps to communicate and deliver insights effectively. As we develop our new insight tools for operational visibility, it is vital that the front-end interface to this system provide dynamic data visualizations that can communicate the insights in a very effective manner. As mentioned earlier, we want to tailor the insights and views to meet the needs of the tool's consumers[7].

For instance, to help design and select a cover for an upcoming Netflix Original TV series, Netflix delves into customers insights by finding out and asking questions like, what colors appeal to customers? Do some covers trend more than others? If so, should the personalized recommendation algorithm automatically change? Looking at three title covers (Figure 2.5) illustrates into a color breakdown chart (Figure 2.6).

Figure 2.5 Sample cover designs for a Netflix Original Series

[7] Mavinkurve, Ranjit, et al. "Improving Netflix's Operational Visibility with Real-Time Insight Tools." The Netflix Tech Blog. Jan 16, 2014. Accessed on June 16, 2015.

According to Figure 2.6, Hemlock Grove and House of Cards contain similarities while Arrested Development does not. These differences can help quantify any "discernable impact on subscriber viewing habits, recommendations, ratings, and the like"[8] With such high production costs, knowing subscriber data is crucial. Television covers aside, what stands out in this example is the customized approach (i.e. aside from bar charts and scatter graphs) for learning about subscriber preferences in an easy way to understand.

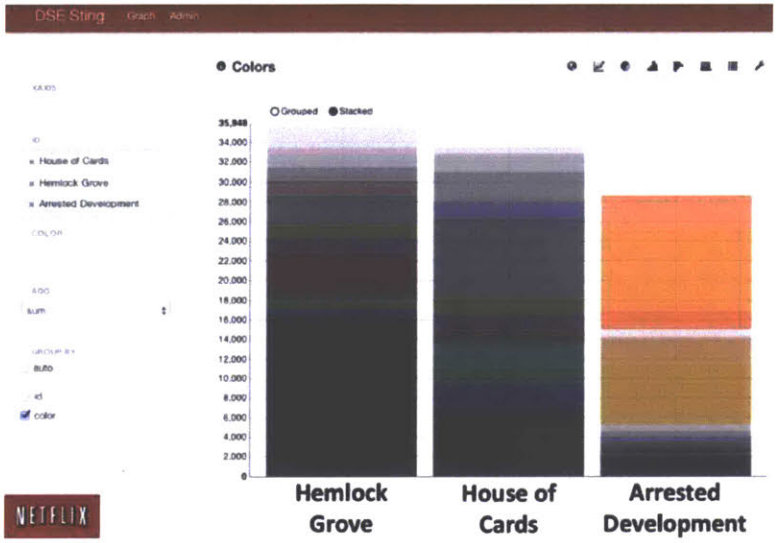


Figure 2.6 Color comparison of different cover designs. Source: techblog.netflix.com

A color comparison creates customer insights that can also inform decisions about the next cover design. Netflix uses the information to see what colors appeal to customers and if customers trend toward specific types of covers for viewing. If so, the Netlifx team had considered whether customer preferences in cover types should change personalized recommendations. Figure 2.3 illustrates the team's visualization goals to help communicate and deliver information effectively:

> As we develop our new insight tools for operational visibility, it is vital that the front-end interface to this system provide dynamic data visualizations that can communicate the insights in a very effective manner...we want to tailor the insights and views to meet the needs of the tool's consumers. We envision the front-end to our new operational insight

[8] Simon, Phil. "Big Data Lessons from Netflix." Wired. March 2014 http://www.wired.com/2014/03/big-data-lessons-netflix/

tool to be an interactive, single-web page application with rich and dynamic data visualizations and dashboards updated in real-time. There are several components within the design: a top level navigation bar to switch between different views in the system, a breadcrumbs component highlighting the selected facets, a main view module (a map in this instance), a key metrics component, a timeline and an incident view, on the right side of the screen. The main view communicates data based on the selected facets (Simon).

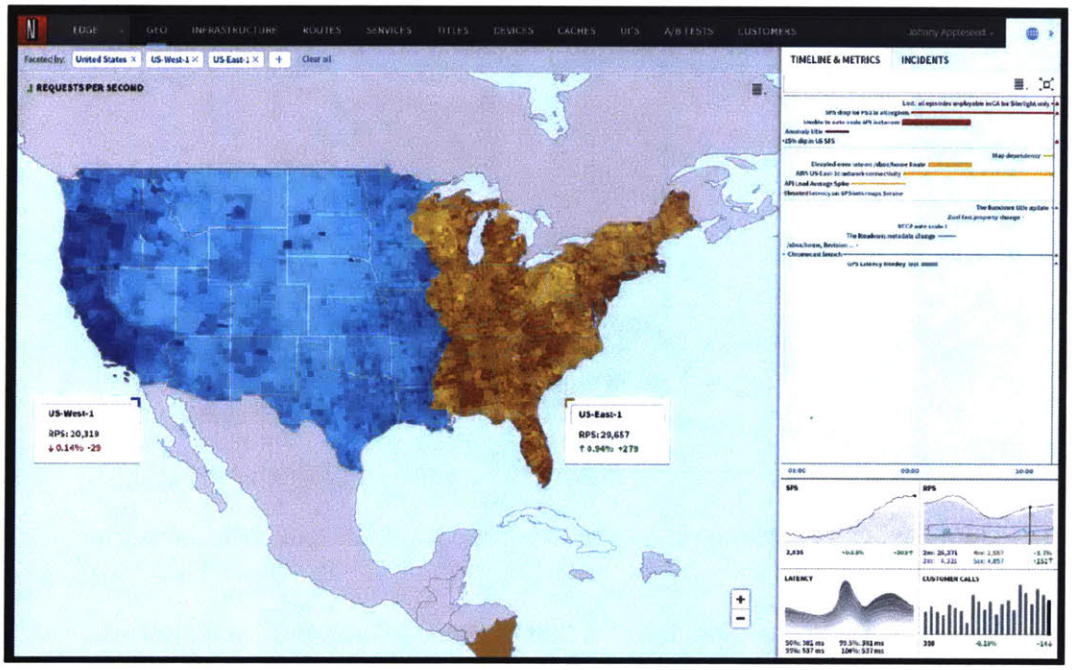


Figure 2.7 (a) Netflix mockup dashboard with dummy data. Source: techblog.netflix.com

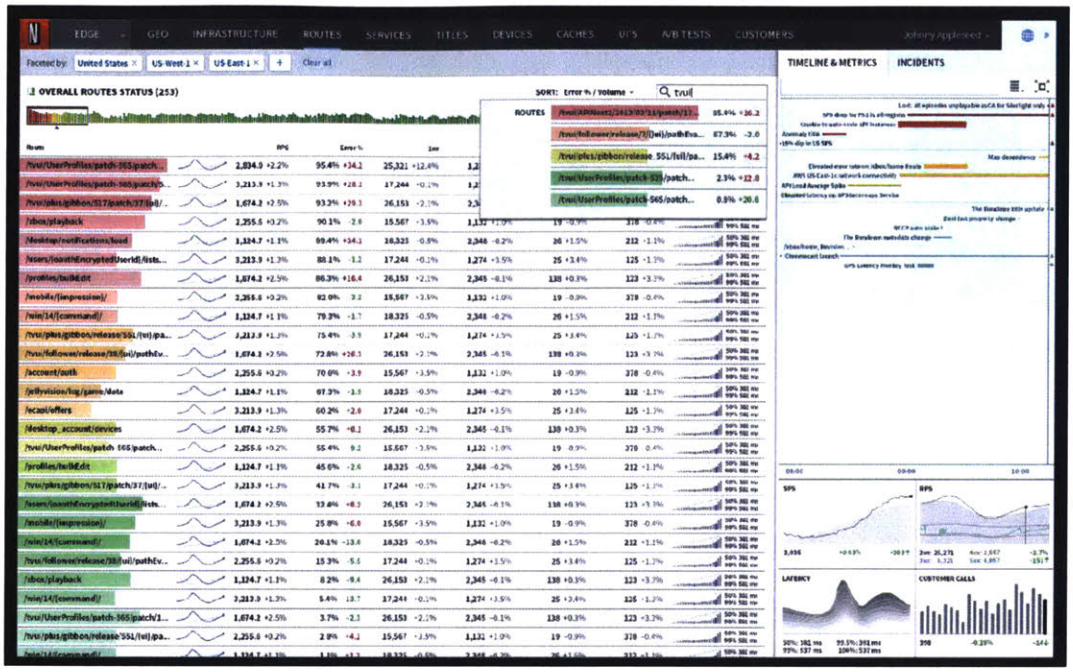


Figure 2.7 (b) Source: techblog.netflix.com

The proposed tool by Netflix illustrates a dashboard with apparent usage of color schemes to immediately identify the overall picture of the system. To illustrate trends, traditional graphs accompany the color coded graphics either in map form or bar chart. In either case, the use of color-coded information and simple traditional plot diagrams are one of the key takeaways from the Netflix example. The next subsection introduces point cloud scanning with LIDAR and image-based 3D modeling at its current state as developed by Autodesk.

2.2.2 Image-based 3D Modeling

Besides 3D geometric and Building Information Modeling software to create 3D models, capturing three-dimensional information in the form of 3D point clouds is brought possible by image-based 3D modeling or LIDAR (laser scanner) scanning of a spatial environment. For best representation and accuracy, a laser scanner measures a large number of 3D coordinates on an object's surface in a very short time (Figure 2.8). The points are then extrapolated to reconstruct a 3D object.

Figure 2.8 (Left) A laser scanning device in MIT's Lobby 7. (Right) A 3D point cloud of Lobby 7 produced by the laser scanner.

While laser scanning offers greater accuracy and is primarily used by professional contractors, its cost has not yet decreased for everyday consumer consumption. The advances in computer vision algorithms, more specifically, multiple view geometry, however, allow for cost-effective methods for capturing 3D point clouds - individuals can use their mobile device to capture images of objects that in turn reconstructs into 3D point clouds, albeit with lesser accuracy than laser scanning. The algorithmic process for doing so relies on the Structure from Motion (SfM) technique.

SfM is a computer vision algorithm to take a sequence of two-dimensional images to create a three-dimensional object relying on Epipolar geometry, or a projection process of multiple view geometry (as discussed in Section 5.2.6). That is, an image is processed by computing similar views between one another. If each camera contains the same point of view, a ray intersects from each camera onto the subject. From the ray, triangulation can be computed between the cameras to determine the distance and location of each pixel in three dimensional space. Figure 2.9 shows Autodesk's 123d Catch software, an image-based 3d modeling tool. The user stands and captures an image of the subject in the center and repeats the process while circumventing the subject. The images undergo Structure from Motion and result in an image-based 3d model. The process and Structure from Motion algorithm is further discussed in Section 5.2.6.

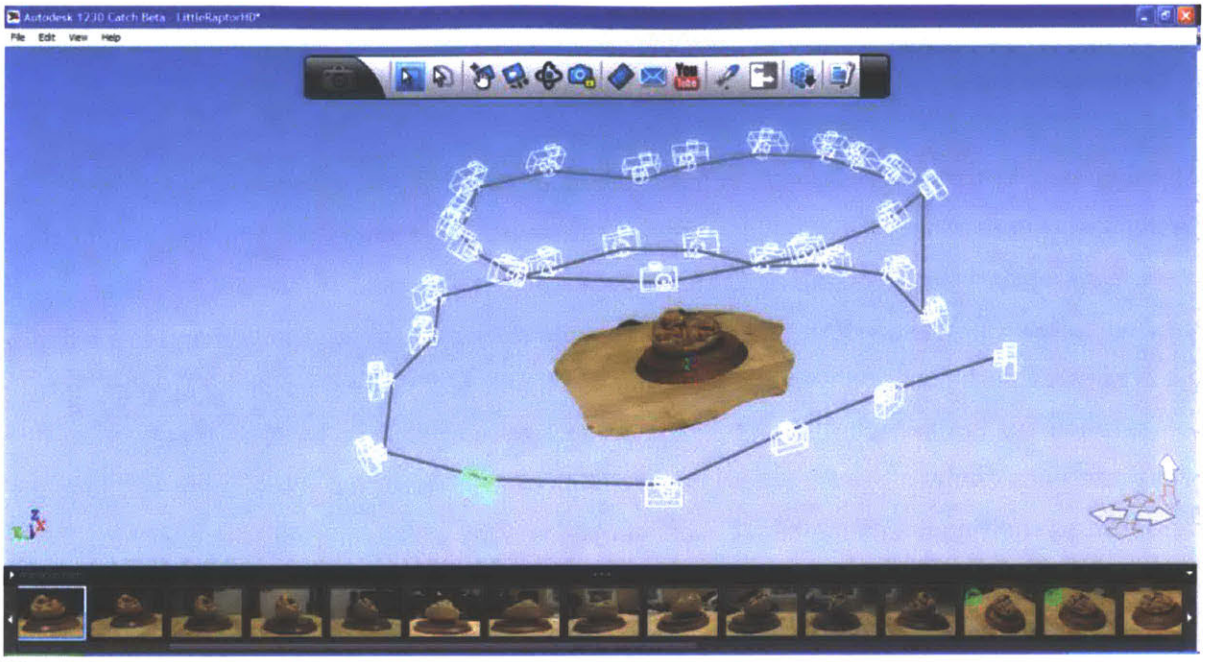


Figure 2.9 Autodesk's 123D Catch photogrammetry desktop software.

In academic research, "Building Rome in a Day" is a project by Professor Noah Snavely and other research team members at the University of Washington. They had developed a high-performance bundle adjustment software, or software for recovering accurate and dense 3D geometry from images that generates an image-based 3D point cloud model of buildings in Rome, Italy (Figure 2.10). They reconstruct entire cities which poses challenges in the 3D reconstruction pipeline, namely large scale optimization.

Figure 2.10 Building Rome in a Day project

2.3 Research Deliverables

The proposed system in this research differentiates itself from the examples in the previous section by 1) data mining publicly uploaded images from the internet about a building, and using the images as a means to create a 3D image-based model, thus offering a 3D model of an existing building in post construction; 2) Analyzing the publicly uploaded images and extracting the original camera positions with the ability to search and filter camera positions by year; 3) A visualization tool of WiFi activity around campus; 4) Text analysis from tags; and lastly, 5) A cross reference analysis between the datasets. Figure 2.11 illustrates some of the software deliverables showing the image-based reconstruction on the web, along with the ability to perform simple linear and volume measurement analysis. Figure 2.12 shows datamined images from the web with their extracted camera positions relative to the photographer's subject. Figure 2.13 shows the accumulated WiFi activity around campus throughout a selected period of time which highlights the population crowd density.

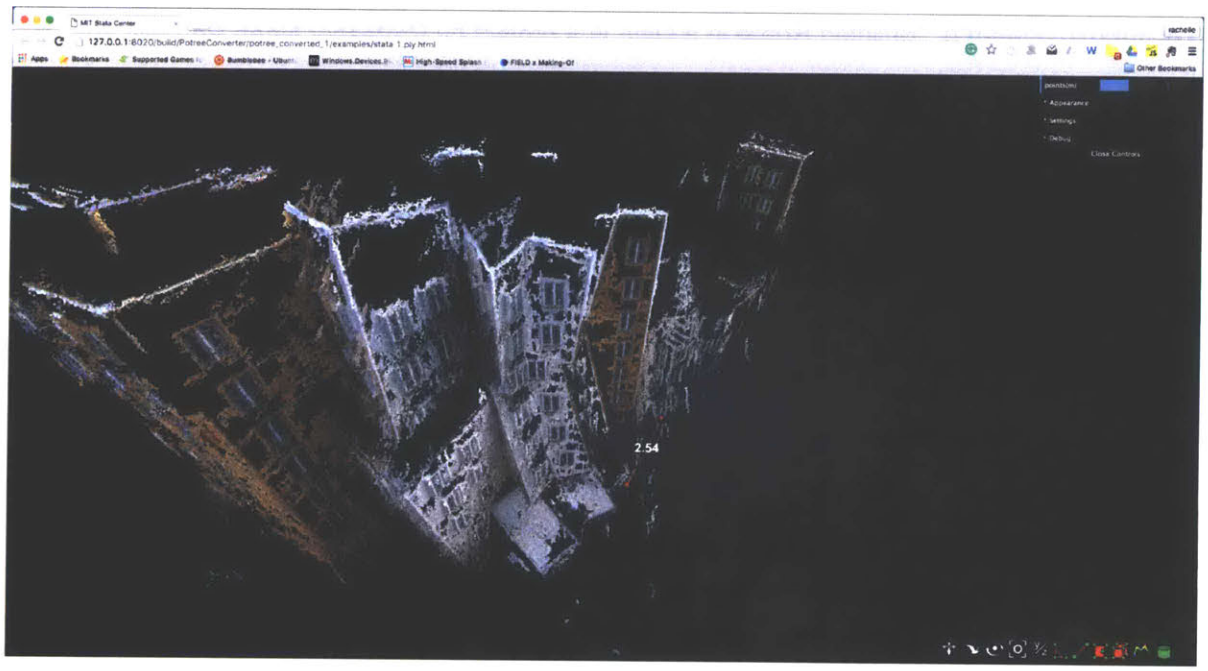


Figure 2.11 (a) 3D reconstruction of publicly uploaded images of Stata Center, some of the features include measuring the linear distance of objects.

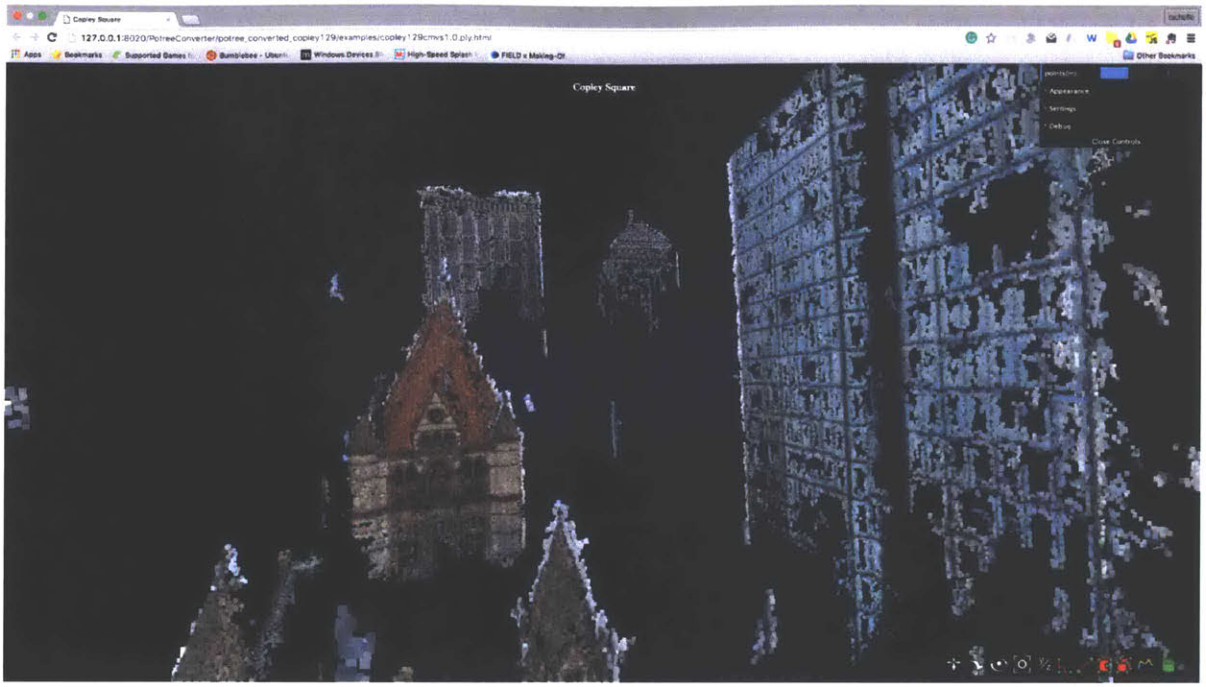


Figure 2.11 (b) 3D reconstruction of Copley Square in Boston.

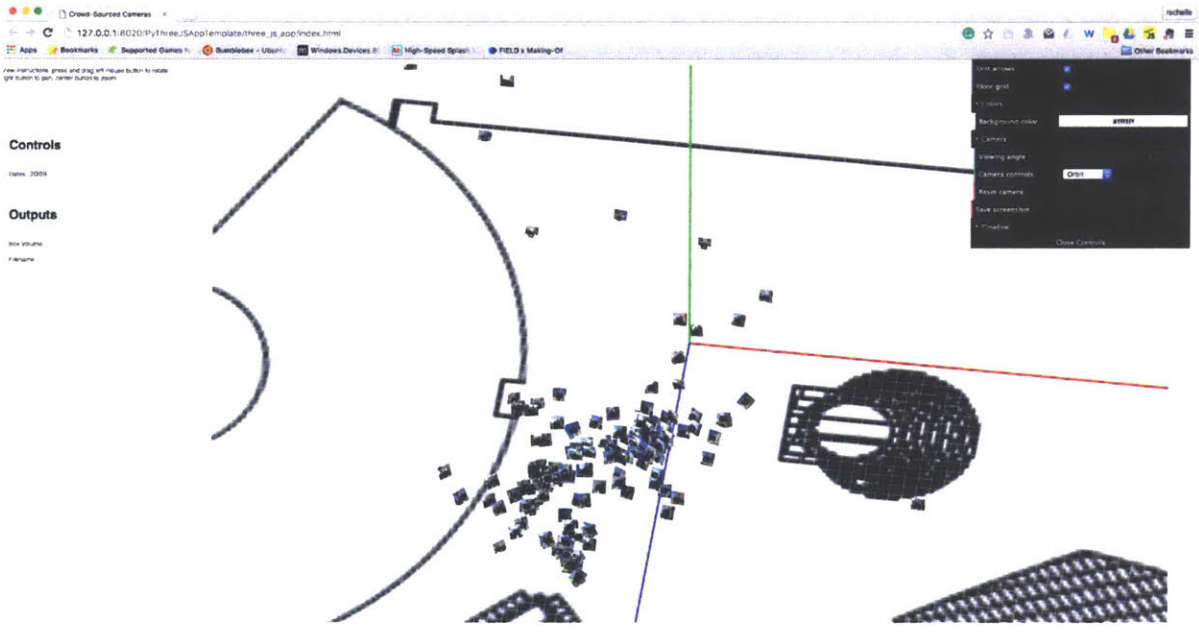


Figure 2.12 (a) A cluster of original camera positions extracted from publicly uploaded images in 2009.

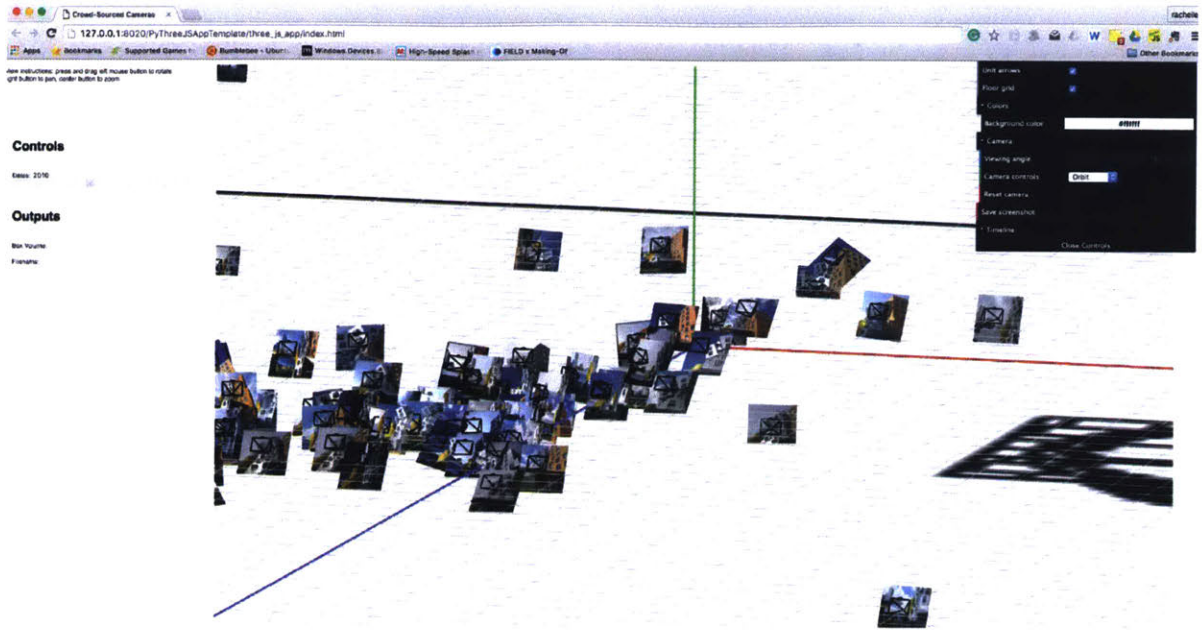


Figure 2.12 (b) Camera frustums in the original rotation, translation, and position accompanied by image thumbnail.

Figure 2.13 A web interface displaying the number of devices connected to a WiFi access point throughout a particular time period.

2.4 User Response

The results and proposal for this research begins a conversation where gathering public and/or private data can inform how we understand the effective or ineffective use of public spaces, whether or not the design of a built environment actually conforms to the original design intention. The tools developed for this research offers facilities management and designers, city management, and historians the ability to test and verify hypotheses for built environments long after construction. The following sections describe feedback and commentary from some of the use case personas.

2.4.1 Facilities Management Feedback and Evaluation

A software application built for this research proves itself useful when actual users *need* a program because the tool fits their goals and hopefully makes their work easier to perform. For the MIT Environment, Health, and Safety (EHS) office, they currently do not have a way to quickly assess the population density of an area on campus other than to send a person to physically be present at major campus events and then perform a head count. With access to the WiFi access point data, mapping the number of connected devices to an access point by latitude and longitude allows the MIT EHS staff member to estimate the number of people connected by their mobile devices in any WiFi enabled area.

Based on discussions with MIT EHS, they find the software application useful for the following features that would allow the staff to:

- Get the maximum number of people at any location at a large and small scale.
- Schedule activities.
- Track visitor/crowd flow during MIT and non-MIT events (i.e. 4th of July).
- To have a diagnostic tool that EHS (and any other institution) do not currently possess because it does not exist.
- Good to know where people gather and don't gather on campus.
- For real-life emergencies, get a fine granularity on the number of individuals in an area.

- Discover how long people exit buildings during fire drills. The current process is done manually by undertaking a head count with a stopwatch, but people do not exit the building from the same door and keeping track of everyone can be challenging.

Other comments from MIT EHS include how the application's usefulness can serve MIT Facilities in terms of resource and emergency management. Additionally, using the data for predictive analytics, or to anticipate when crowds will gather at a campus location is another distinctive attribute that MIT EHS would need at their disposal. MIT EHS also mentioned that the use of images would help validate the information received via dispatch. Their main goal is to be able to quickly assess the number of people in any area on MIT's campus at different levels of granularity.

The software application presented to EHS was based on static data from September - October 2014 and was cleaned for production. Another version is currently in progress, this time to visualize and analyze near real-time data at the request of EHS staff for use in emergency management at MIT.

2.4.2 A Historian's Feedback and Evaluation

Upon discussing the results of this research with Professor Howard Burns, a visiting MIT Professor and former Professor in the History of Architecture at Harvard University, he mentioned how "useful this research is for historians [in terms of extracting the original camera pose and seeing what the user frames in their photographs]" and referred to how the 18th century Italian artist, Giovanni Battista Piranesi, created perspective drawings of architecture that influenced architects and architectural works during the Italian Renaissance (Figure 2.14). In a similar manner, learning from the eye of the public about the built environment can play a role in how architects and city planners design today (Figure 2.15).

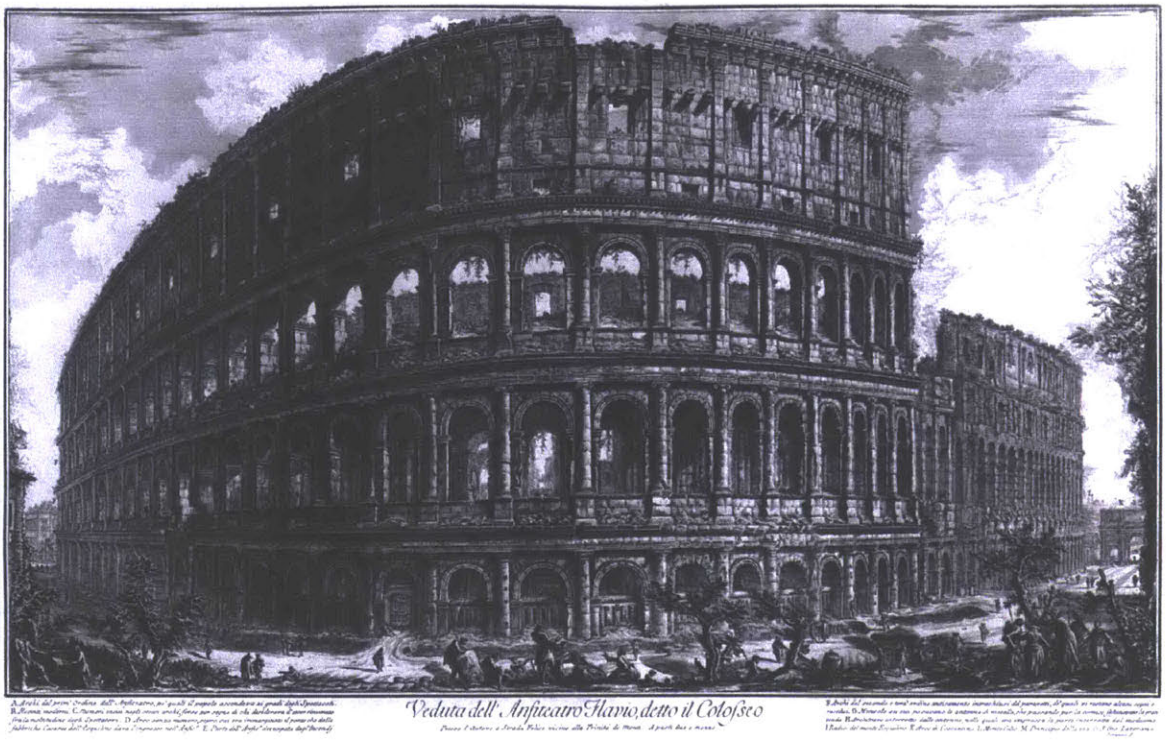


Figure 2.14 The Colosseum[9], etching, 1757.

[9] By Giovanni Battista Piranesi - R.S. Johnson fine art, Public Domain, https://commons.wikimedia.org/w/index.php?curid=5765775

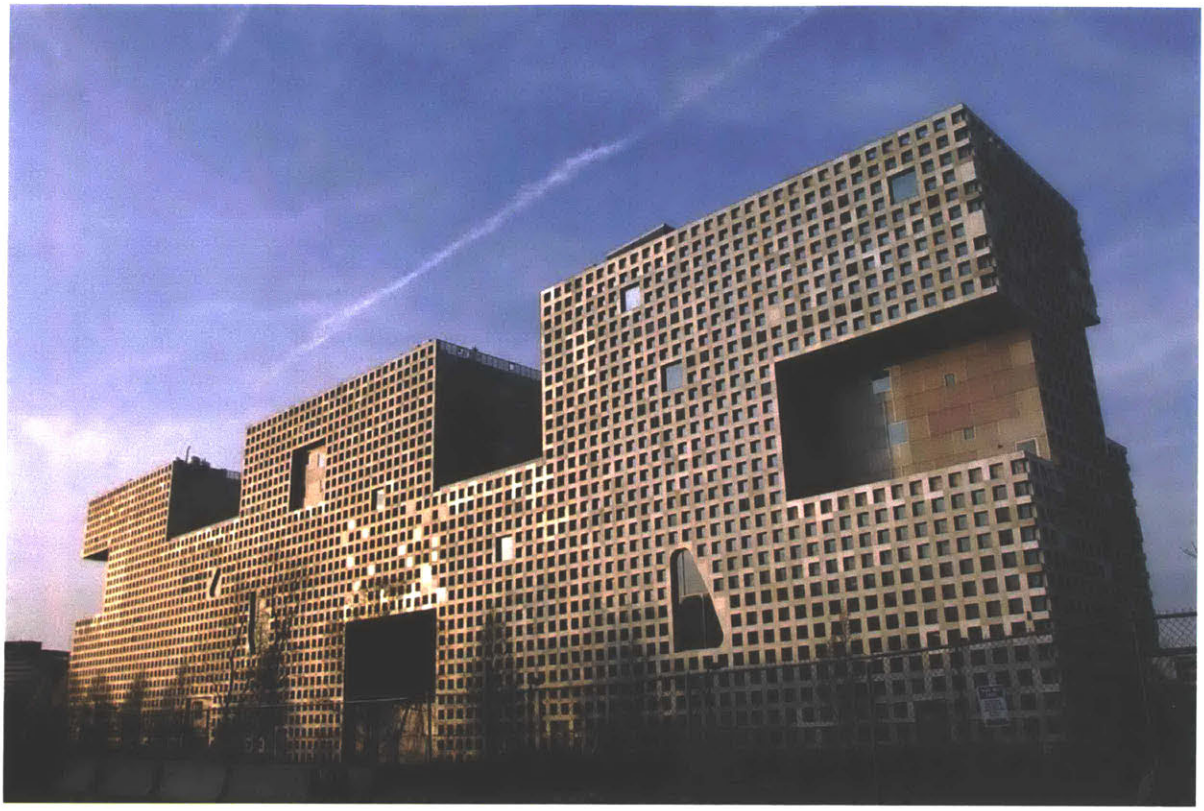


Figure 2.15 Simmons Hall, publicly uploaded photo.

Analyzing many publicly uploaded photos provide building managers, designers, historians, and city management the ability to see what the general community finds important or iconic about places of interest -- thereby allowing future designs to accommodate new criteria based on community data.

Chapter 3

Exploratory Data Analysis

3.1 Introduction

This research relies on data mining and extrapolating information from three public and private web-based data sources: photographs from Flickr, an online repository of user-uploaded public photos; MIT WiFi data gathered from the servers of the MIT Information Systems and Technology (IS&T) Department; and public text from the accompanying captions on gathered photographs. The indoor and outdoor spaces of MIT's Stata Center building, Simmons Hall dormitory, and Boston's Copley Square, all serve as the main subjects for data mining and analysis. A key characteristic with the image, WiFi, and text datasets are the methods of unique exploratory techniques for describing and summarizing each dataset. Additionally, where something is recorded is of primary interest, the datasets herein share the component of location and time of when and where the record had occurred. Do people photograph Stata Center more in the Spring or Summer time? Is there a seasonal pattern? Can image clusters reveal why individuals prefer to photograph one building facade versus another? What is the likelihood that people will occupy classroom 34-144 versus Building 34-123 at 2pm on a Thursday? How many pictures are taken at any given year? Is there a trend to taking pictures of

Stata Center or Copley Square? This chapter interprets, analyzes, creates assumptions, and sets alternative perspectives in order to answer the questions above. Table 3.1 summarizes the buildings explored and the corresponding datasets available for this research.

Building	Image (EXIF)	Text	WiFi
Stata Center	O	O	O
Simmons Hall	O	O	O
Copley Square	O	O	X

O = Available

X = Unavailable

Table 3.1 Buildings and datasets explored.

3.2 Descriptive Analysis of Stata Center through WiFi Access Points and Crowd-Sourced Images

When a picture is taken, a digital camera tags each photo with hidden information, or EXIF (Exchangeable Image File Format) metadata. Information such as the date and time when the photo was taken, shutter speed, lens aperture, focal length, make and model of the camera, and sometimes geolocation along with other camera parameters consist of EXIF data. Extracting the cameras make and model from a photograph can reveal the population of individuals taking exterior images of Stata Center. Figure 3.1 shows the camera types that people use to photograph the building. Nearly 500 images are taken with a Canon camera, while Nikon takes second place, and Apple iPhone's take third. Furthermore, Figure 3.2 lists a sample of camera makes and associated models which reveals that the majority of individuals photograph with DSLR cameras such as the Nikon D Series and Canon EOS series. What does this information say about the type of individuals documenting Stata Center? That the images are rarely taken by casual users with a mobile phone device? That individuals with professional cameras are purposely seeking out geometrically interesting subjects to photograph than a casual user? We can also assume that users with professional cameras probably spend more time composing a photo and experiment with the camera's variety of settings.

3.2.1 What Types of Devices Photograph MIT?

At any rate, the majority of pictures reveal that their owners create photographs with professional type cameras that require a level of customization and creative control. It's important to keep in mind that the dataset only represents those who have taken pictures of Stata Center and uploaded to Flickr, and does not account for the individuals who have decided to also photograph the building but either uploaded to another social networking site or not at all. The pictures data mined for this research represents a sample in time from a database that is frequently receiving new images from users beyond the date that the random sample of images were data mined. The Stata Center local image dataset, for example, has its most recent image from May 2015.

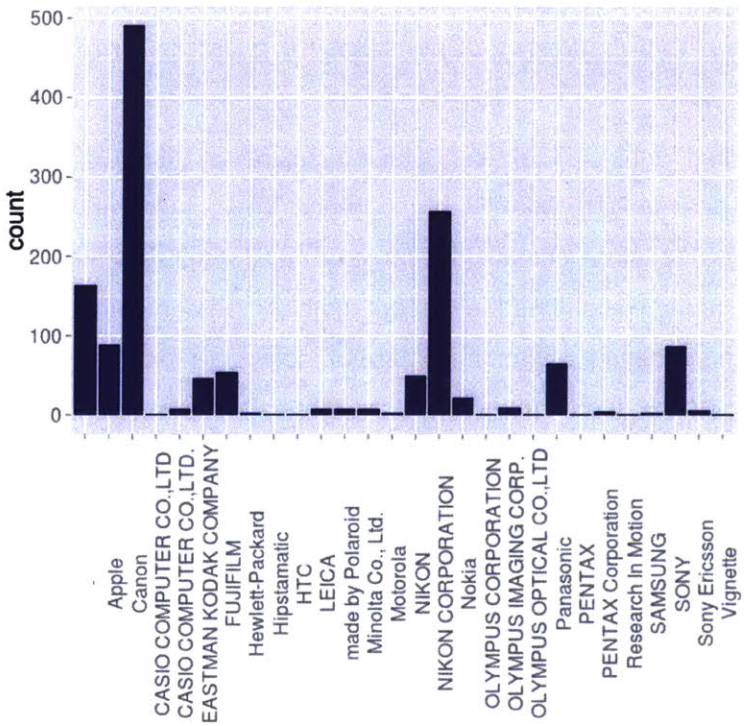


Figure 3.1 A count of camera types used to photograph Stata Center as extracted from each image's EXIF tag.

```
 [1] "Apple\tiPhone 3GS"                                  "Canon\tCanon PowerShot SD1000"
 [3] "Panasonic\tDMC-ZS8"                                 "made by Polaroid\t6M Digital Camera"
 [5] "NIKON CORPORATION\tNIKON D7000"                     "-\t-"
 [7] "Canon\tCanon EOS DIGITAL REBEL XT"                  "Canon\tCanon DIGITAL IXUS 80 IS"
 [9] "Panasonic\tDMC-LZ2"                                 "Canon\tCanon EOS DIGITAL REBEL XTi"
[11] "Panasonic\tDMC-ZS3"                                 "NIKON CORPORATION\tNIKON D200"
[13] "Panasonic\tDMC-ZS8"                                 "EASTMAN KODAK COMPANY\tKODAK Z740 ZOOM DIGITAL CAMERA"
[15] "Canon\tCanon EOS 5D"                                "Apple\tiPhone"
[17] "NIKON CORPORATION\tNIKON D90"                       "FUJIFILM\tFinePix A600"
[19] "Canon\tCanon EOS 40D"                               "Apple\tiPhone 3GS"
[21] "NIKON CORPORATION\tNIKON D40X"                      "NIKON\tCOOLPIX S9"
[23] "Canon\tCanon EOS 7D"                                "NIKON CORPORATION\tNIKON D40X"
[25] "SONY\tDSLR-A230"                                    "made by Polaroid\t6M Digital Camera"
[27] "-\t-"                                               "-\t-"
[29] "NIKON CORPORATION\tNIKON D80"                       "SONY\tDSC-P10"
[31] "PENTAX Corporation\tPENTAX K10D"                    "Canon\tCanon EOS 40D"
[33] "Nokia\tN97"                                         "Canon\tCanon PowerShot S95"
[35] "Canon\tCanon EOS DIGITAL REBEL XTi"                 "NIKON CORPORATION\tNIKON D70"
[37] "Canon\tCanon DIGITAL IXUS 55"                       "EASTMAN KODAK COMPANY\tKODAK Z812 IS ZOOM DIGITAL CAMERA"
```

Figure 3.2 A list of camera makes and models used.

3.2.2 Raw and Relative Frequency of Documenting Stata Center

Based on Table 3.2 and Figure 3.3, a trend of taking pictures of Stata Center increases since 2002 and reaches a peak in 2010 with 280 photographs. This raw frequency of 280 photographs, however, does not indicate how large or small the 280 photographs are. In that case, the relative frequency describes the percentage or proportion of individuals who took photographs within the sample population, so 22% of individuals in the sample population of 1,230. It's interesting to note that in the years of the Great Recession during 2007-2009, this is when pictures of Stata Center began an upward trend as Boston experienced a decline in tourism[10]. The data helps to ask certain questions: Did individuals have more time to spare and dedicate to photography? This research does not answer such questions, but speculates the possibilities for trends evidenced in the data.

[10] "Boston's People and Economy." Vol. 1, 2009. Published by the City of Boston. http://www.cityofboston.gov/

Year	Raw Frequency	Relative frequency
2002	8	0.006504065
2003	6	0.004878049
2004	2	0.001626016
2005	4	0.003252033
2006	111	0.0902439
2007	168	0.1365854
2008	175	0.1422764
2009	180	0.1463415
2010	280	0.2276423
2011	55	0.04471545
2012	124	0.100813
2013	66	0.05365854
2014	45	0.03658537
2015	6	0.004878049

Table 3.2 Raw and relative frequency of Images taken of Stata Center since 2002

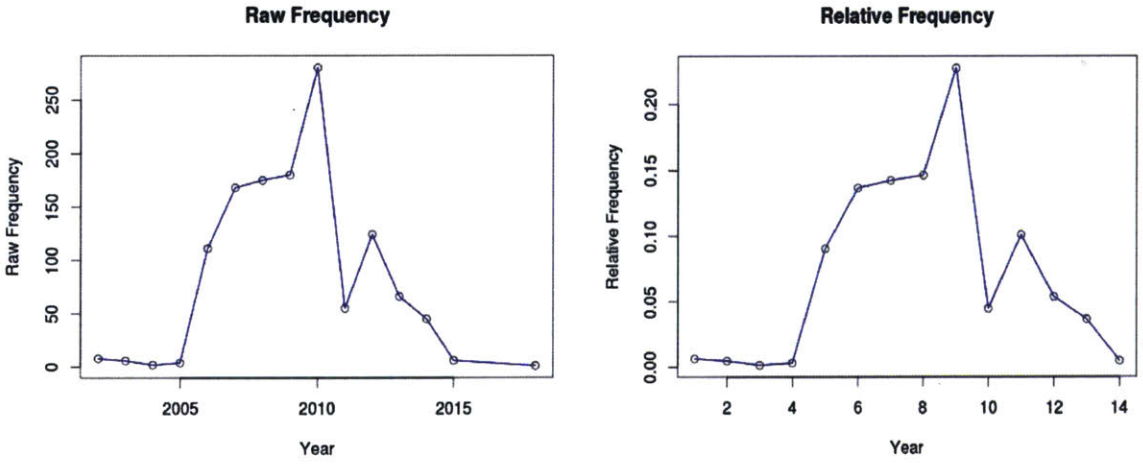
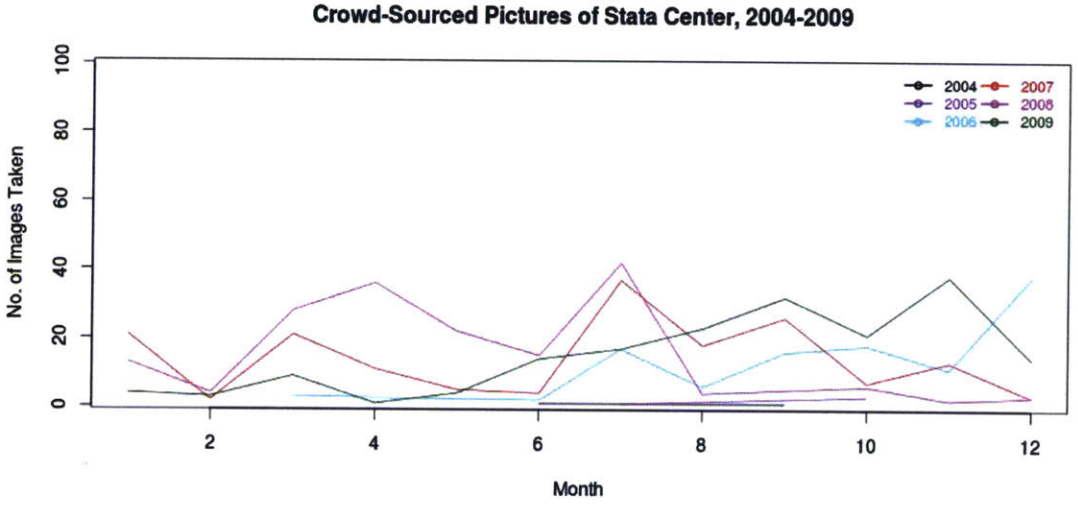


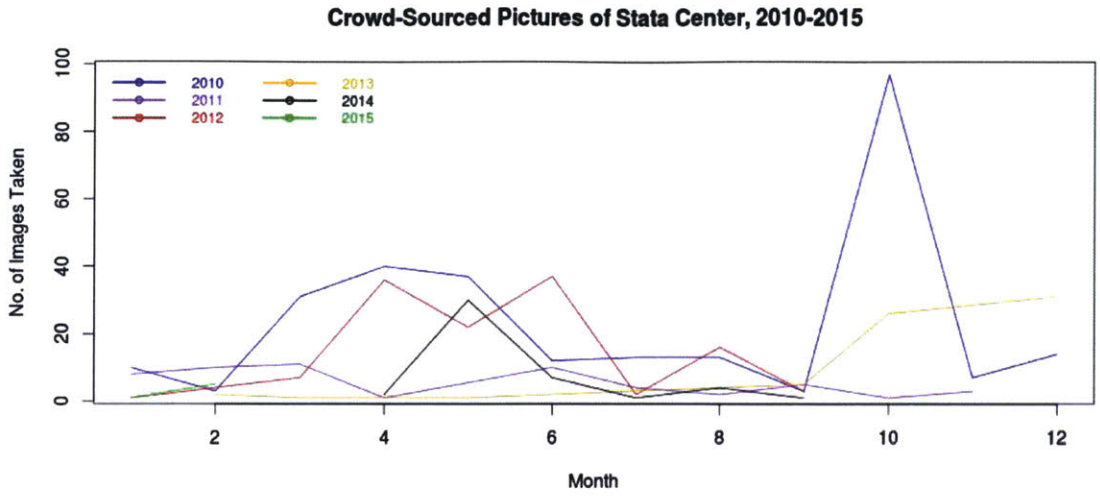
Figure 3.3 Raw and relative frequency plot of photos taken at Stata Center since 2002.

3.2.3 Stata Center in Time Series

Looking further, Figure 3.4 breaks down the number of pictures taken by year and month. From the year 2004 to 2009, the number of pictures taken of Stata Center is relatively low in the winter months of December to February, then begins to increase around spring time and slowly declines until summer begins. Stata Center in the summer months of June to August experience the most pictures taken of itself until the trend declines towards the Fall season with a much smaller peak. With the full effect of the four seasons in New England, weather appears to play a significant role in the spikes and troughs for each year's trend line. The seasonal pattern repeats itself with some similarity for the years 2010-2015, the exception are the summer months which is much lower than the previous graph for the years 2004-2009.



(a)



(b)

Figure 3.4 (a) Number of pictures taken by year and month from 2004-2009. (b) Images from 2010-2015.

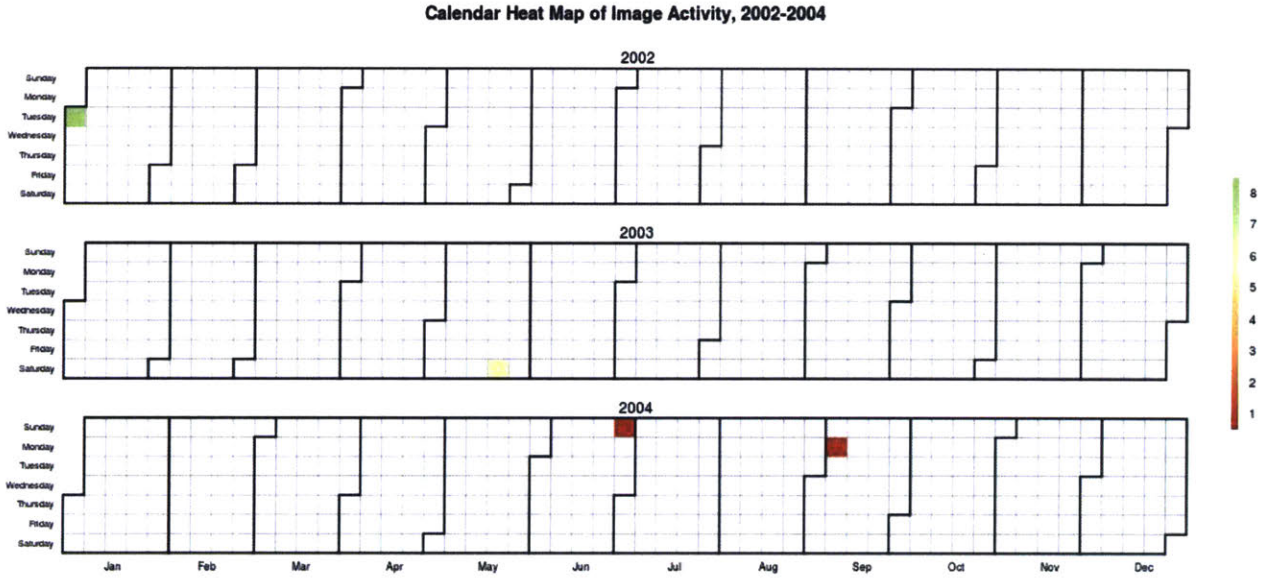
Stata Center opened to the public in 2004, curiously enough there were images taken prior to the opening day. As Figure 3.4a shows, photos between 2002-2004 are thinly dispersed across that time frame in contrast to the years after 2004 with a minimum of one photo taken and a maximum of 8 photos on September 6, 2004. Figure 3.4b shows how Stata Center is becoming quite popular amongst photographers as the years progress. The year 2007 is when Stata begins to pick up traffic and the weekends of July and August become popular months and days for a photo shoot. The trend continues with an average amount of 3 pictures taken in a day per month in 2007 and a maximum of 12 pictures taken on one day during that entire year. As for what days people visit Stata Center the most for a photograph, Table 3.3 shows the number of pictures taken for any given day throughout 2007. Figure 3.5 presents a heat map of all the image activities within Stata Center.

Day of Week	Image Count
Sun	33
Mon	23
Tues	15
Wed	14

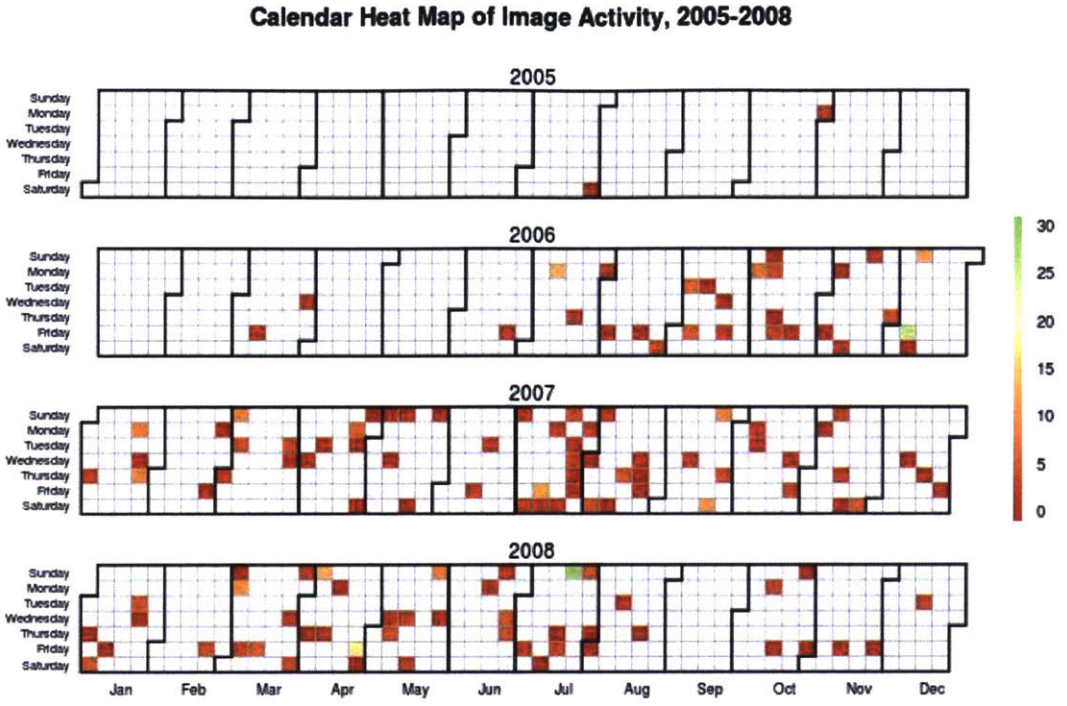
Thurs	29
Fri	21
Sat	33

Table 3.3 Image count for the year 2007.

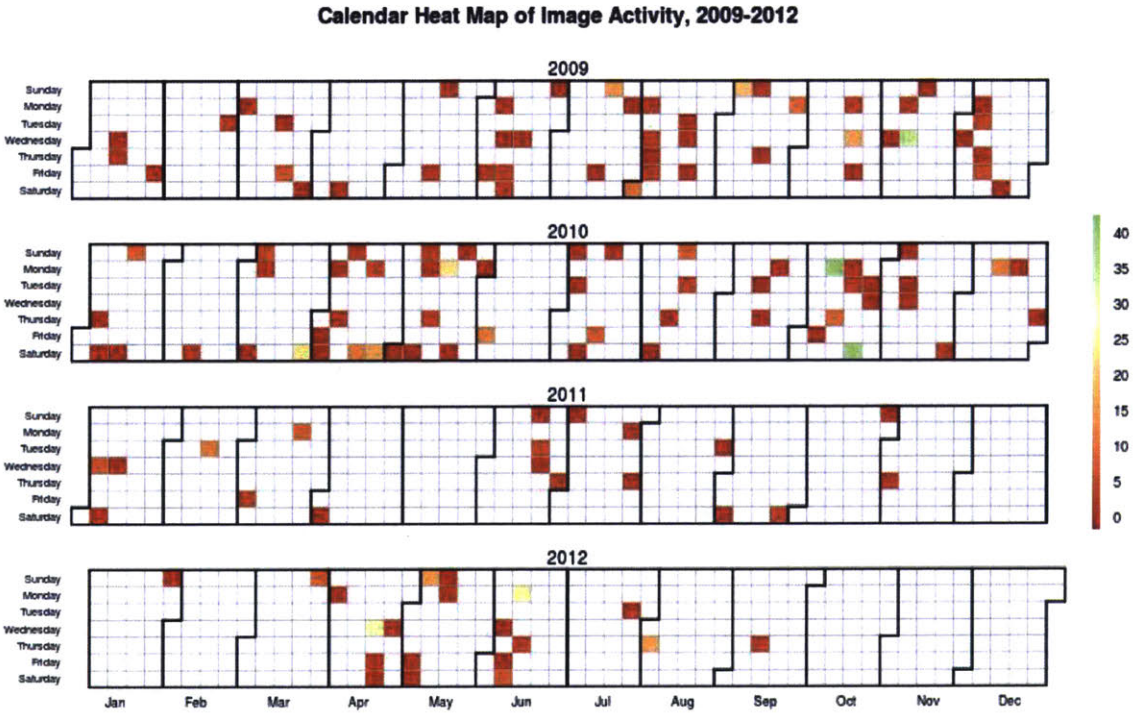
Both Sundays and Saturdays are the most frequented days throughout the week and declines slowly by Monday, then begins to pick up again on Thursdays up until the start of the weekend. The years of 2008 throughout 2010 contains a steady trend with an average number of pictures taken in those years respectively, then noticeably declines by the year 2011 onward.



(a)


Calendar Heat Map of Image Activity, 2005-2008
2005
2006
2007
2008
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
30
25
20
15
10
5
0

(b)


Calendar Heat Map of Image Activity, 2009-2012
2009
2010
2011
2012
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
40
35
30
25
20
15
10
5
0

(c)

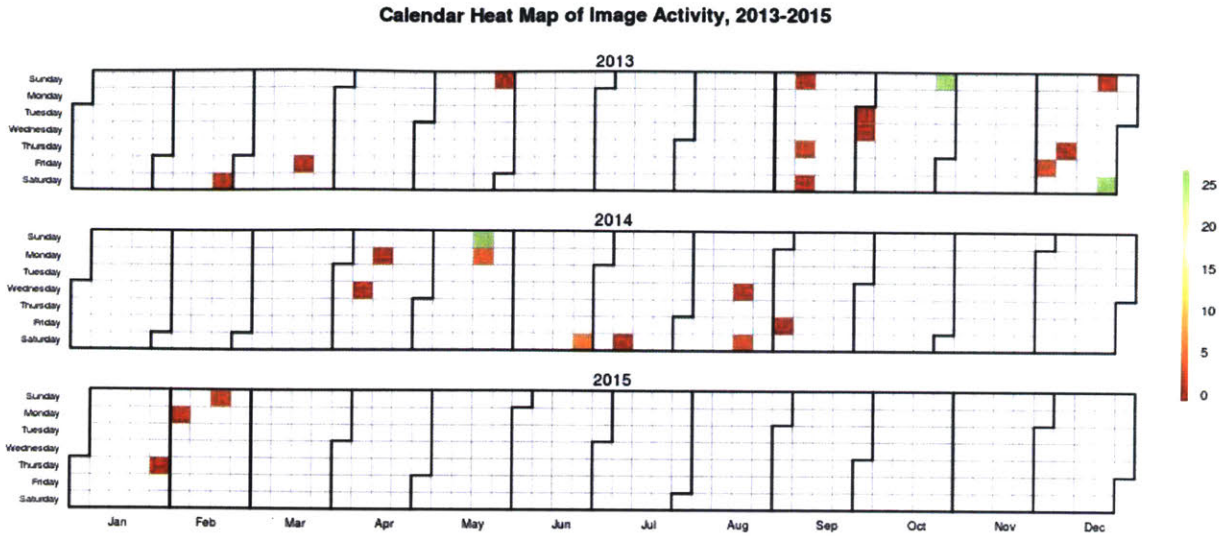


Figure 3.5 Calendar heat map of image activities from the year 2002-2015 as seen in Figures a, b, and c.

Figure 3.6 (a) The image dataset from the web.

Figure 3.6 (b) Gathering the crowd-sourced images to create a 3D point cloud model. The model is an accumulation of images since the year 2002.

3.2.4 Spatial Point Pattern Analysis of Stata Center

This section introduces the spatial distribution of cameras that were used to take pictures around Stata Center (Figures 3.6a and 3.6b). The distribution does not follow a regular pattern, but a pattern of clusters labeled as regions A, B, and C that appear around the building since the year 2002. This pattern repeatedly appears in those regions over time and rarely in others. The formation of a cluster typically occurs when there is an attraction[11]. One side of the building's facade is valued in some way and/or an accessible path to that area makes it convenient for people to enter and pause for a picture (Figure 3.7). If the latter were true, however, then why don't the majority of people photograph other facades? With the construction of the Sean Collier Memorial sculpture in Spring 2015 (seen to the right of region A in Figure 3.7), time and a new image dataset beyond the one used for this research will tell if the sculpture has created a new attraction point and spawned the formation of yet another cluster.

[11] Bivand S. Roger, et. all. Applied Spatial Data Analysis with R. Springer, New York: 2015.

Figures 3.8 and 3.9 shows a screenshot of the custom research tool developed to plot the extracted camera positions from photographs.

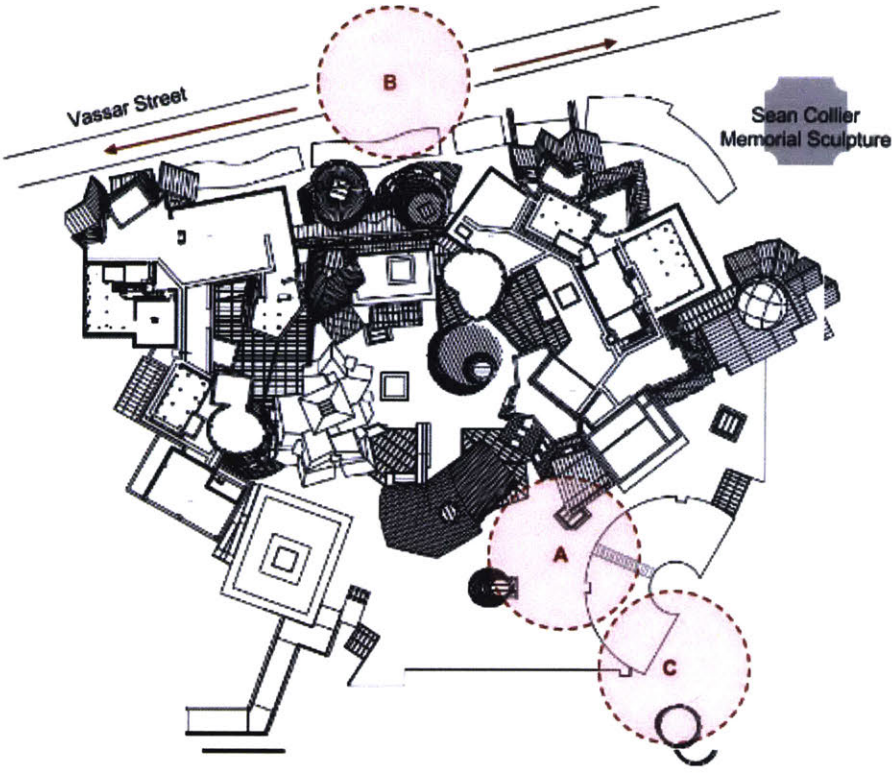


Figure 3.7 Camera clusters that occur around Stata Center since the year 2002.

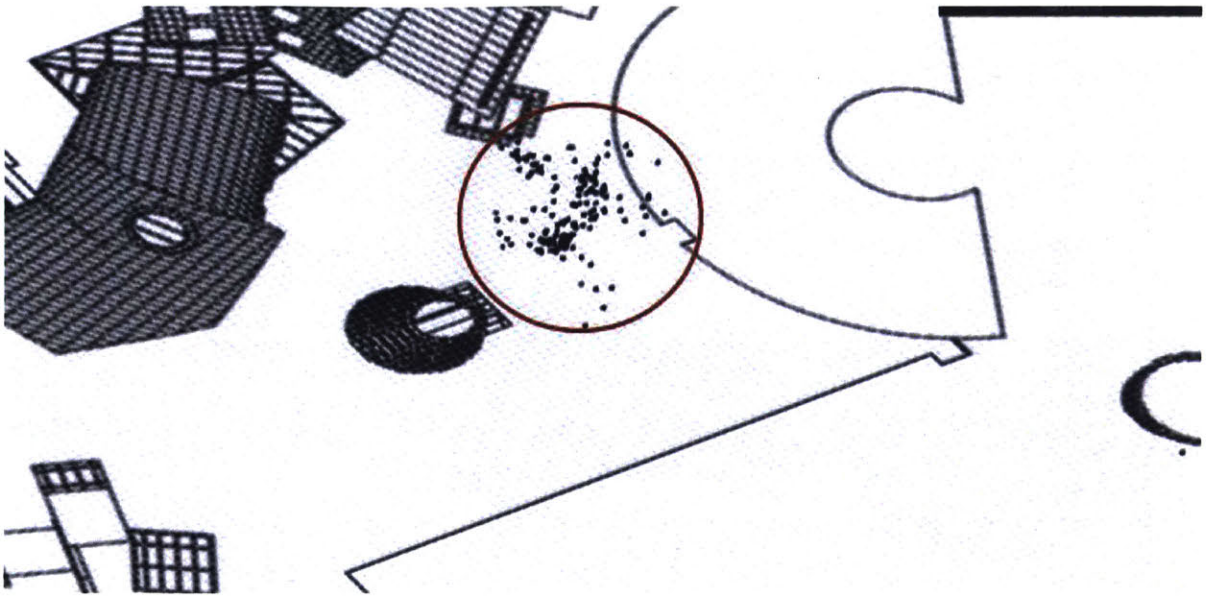


Figure 3.8 A roof plan of Stata Center and the calculated camera positions from 2D images taken by the general public since 2002.

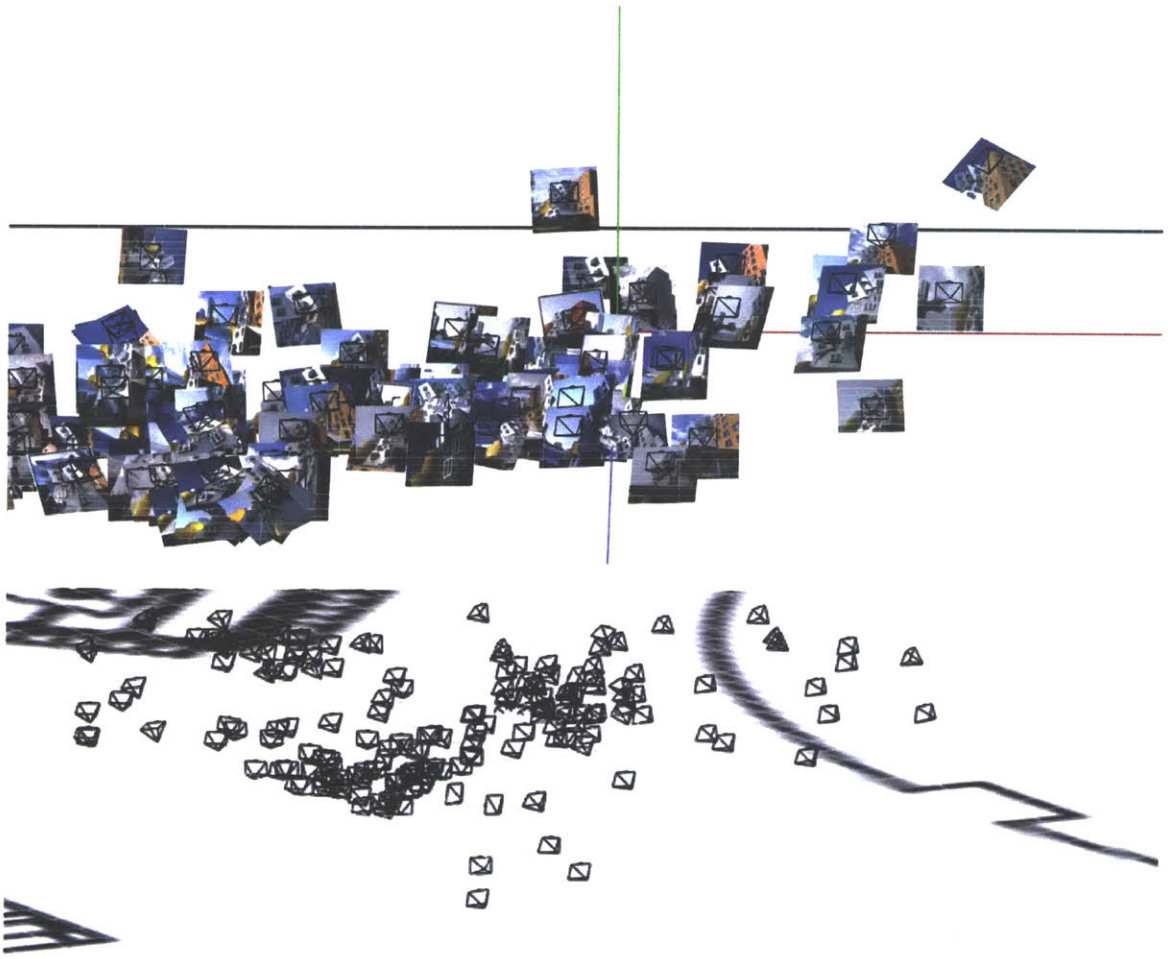


Figure 3.9 Cluster of camera objects and original images shown in the calculated rotation, translation, and position of each shot taken by a user.

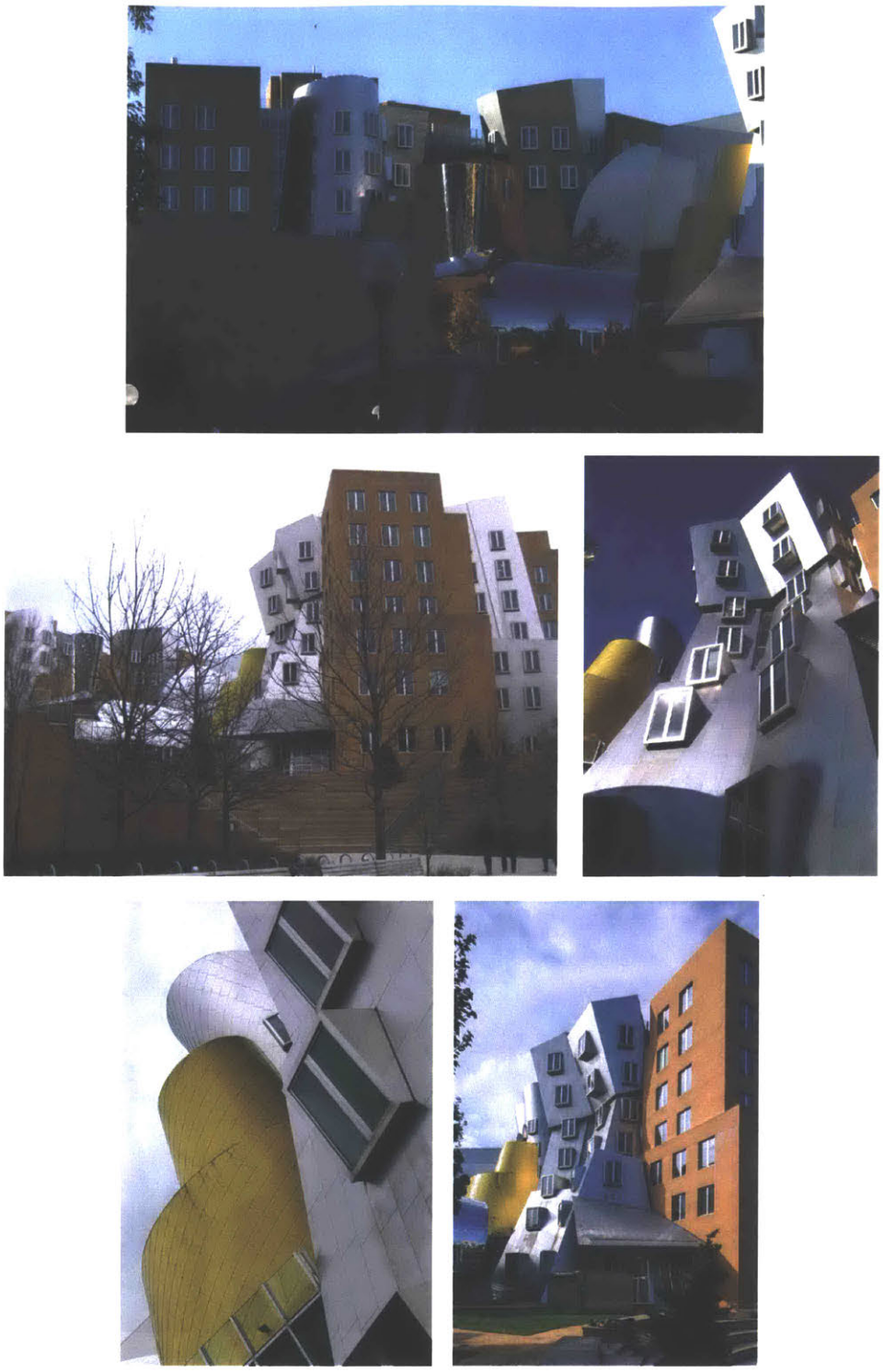


Figure 3.10 (a) A cluster of photographs were taken in region "A" and "C" of Stata Center.

(b). Images seen in Cluster B

The visualization of clustered camera frustums demonstrates a qualitative way of acknowledging that individuals gather at particular locations to photograph Stata Center. By extracting the translation coordinates (X, Y, Z) of each camera location, we can confirm the number of cluster formations that occur around the building with a quantitative approach. Given a set of observations (camera coordinates), the aim is to partition the *n* observations into clusters, *k*, such that $(k < n)\ S = \{S_1, S_2, ..S_k\}$ where S_i are the cluster sum of squares (see Chapter 5.2.4 on clustering optimization problems solved using K-means clustering, an unsupervised machine learning algorithm). The fundamental question is how to determine the value, *k*, in order to create the number of clusters. The "elbow criterion" is used to determine the optimal value of *k,* where the sum of squares are low and increasing k would start to have diminishing returns, in this case, the optimal number of clusters is *k* = 3 (Figure 3.11).

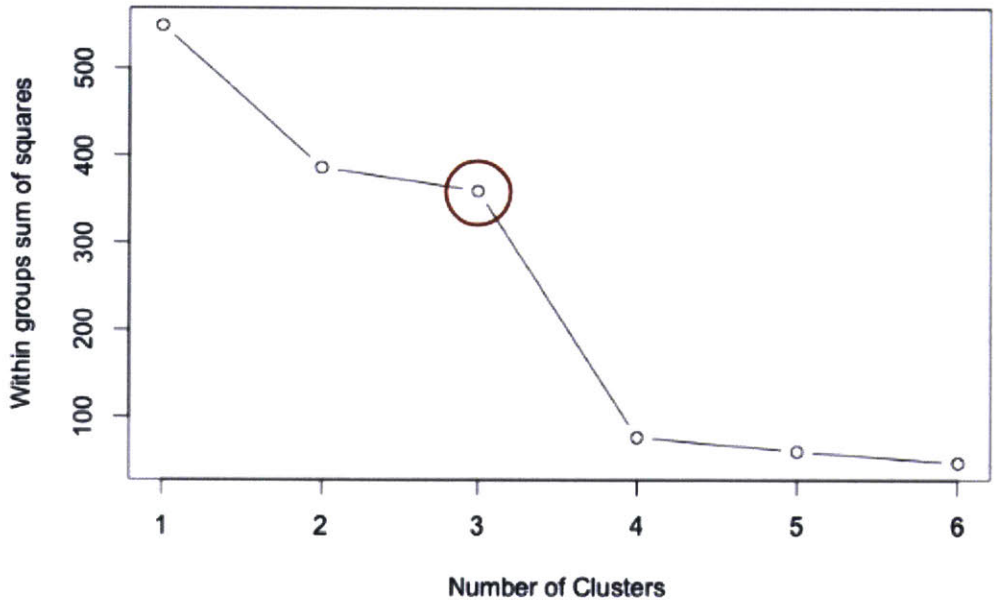


Figure 3.11 Determining the optimal number of clusters, k.

Using k=3, Figure 3.12 confirms the qualitative observation of clusters formed around Stata Center into a quantitative one, where each data point falls into a group formation. With the three cluster formations confirmed, forming observations about the particular dataset can now become a discussion for answering where and possibly why individuals photograph the building from within the three clusters since the year 2004.

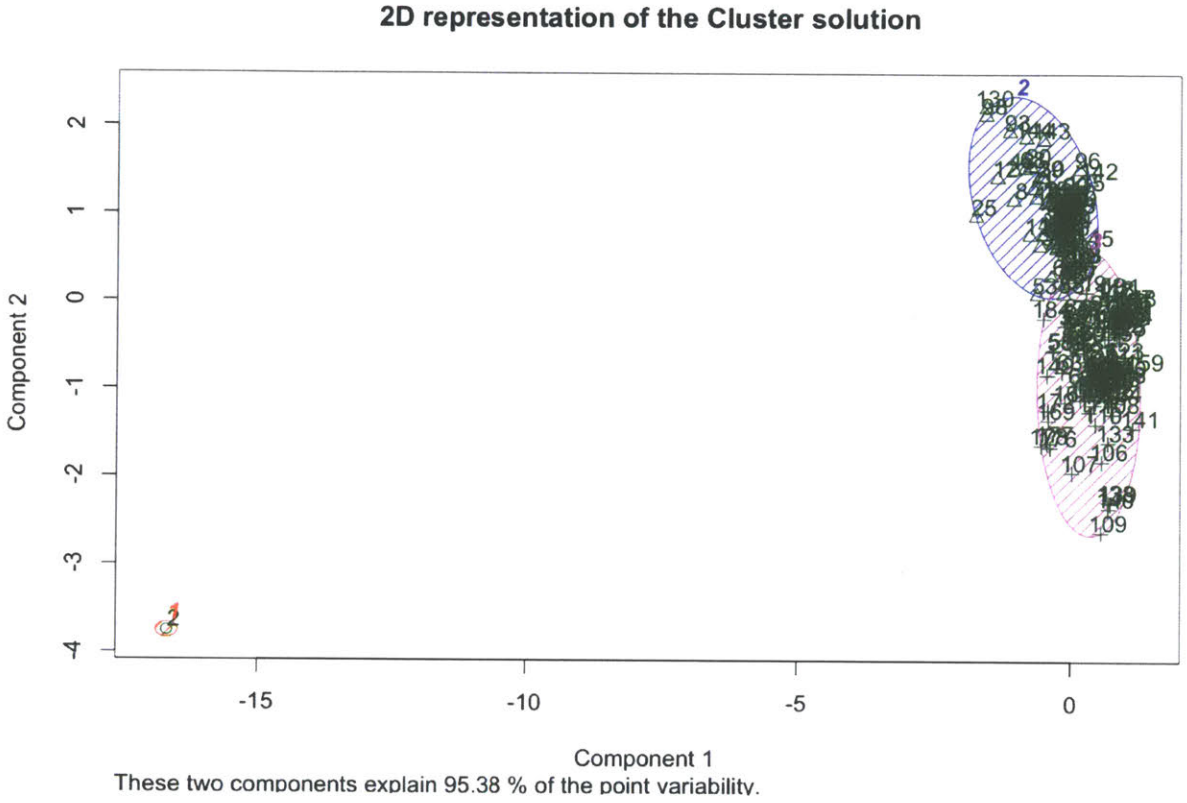
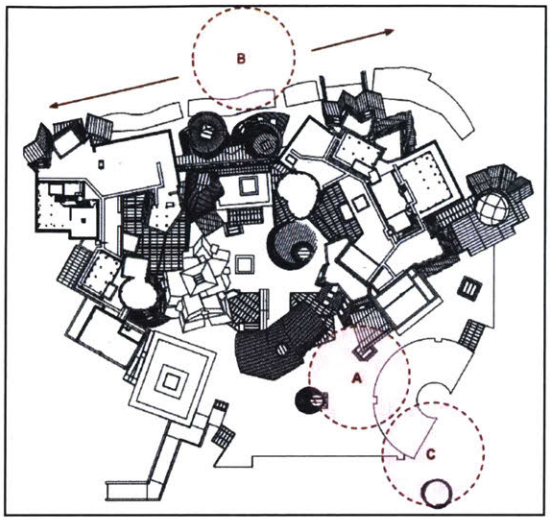


Figure 3.12 Cluster solution.

Individuals tend to climb the stairs to photograph the building in cluster A as illustrated in Figure 3.7, stand near the outdoor auditorium in Cluster C, or pause anywhere along the stretch of Stata Center on Vassar Street as seen in Cluster B. Figure 3.10 displays the photographs taken in the aforementioned clusters Cluster A and/or C is a prime location for the MIT Information Office to construct information signage or promote events to targeted visitors, for example. MIT Facilities can focus their maintenance efforts toward "photogenic" sections of a building which attracts high pedestrian traffic. For a historian, this area confirms that over the passage of time, the majority of individuals had paused to record certain parts of the building while selectively omitting others. City officials would find this information useful so that they can determine

tourism flows and discover what city icons do tourists like to keep as memories -- helping to target and allocate spending on iconic landmarks.

Notice the cluster formations around Stata again, Figure 3.13 shows a map of a green space/courtyard adjacent to Stata Center. Despite the greenery and the number of benches and picnic tables lining the west and north side of the courtyard, the majority of individuals do not take photographs from the grassy area. Typically, a grassy courtyard is an invitation for individual leisure -- holding picnics, casual sports, gathering of friends, reading, and other activities. Perhaps the community uses the grassy courtyard but not the photographers whose intent is to photograph the building. Individuals, then, have a strong preference to capture Stata Center from very specific locations.



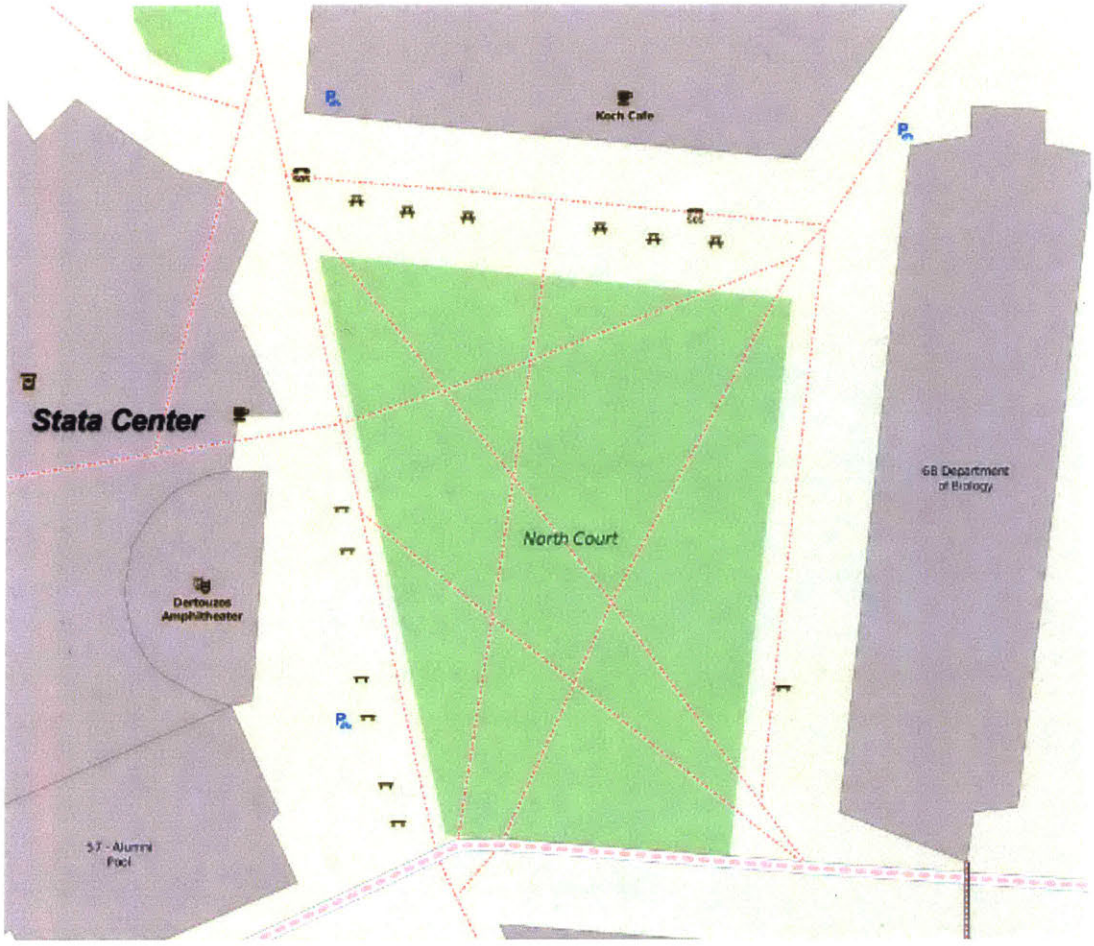


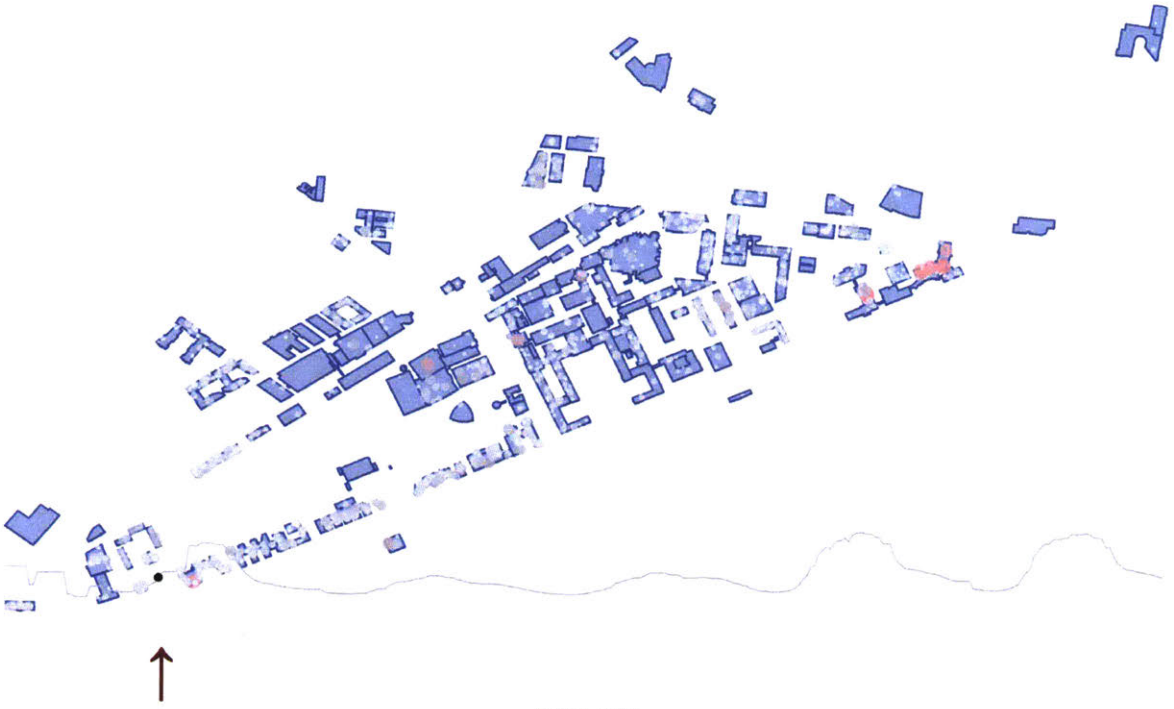
Figure 3.13 Courtyard with benches and picnic tables around the west and north side of the greenspace.

This section of Chapter 3 analyzes data points from the derived camera coordinates from outdoor public images of Stata Center. The next section focuses on indoor data points by analyzing the number of connected devices to MIT's WiFi access points, offering a glimpse at the indoor vs. outdoor relationships within the building and surrounding areas.

3.3. MIT through WiFi Access Points

Each WiFi access point around MIT's campus, records the number of connected devices to itself along with a timestamp. Extracting the recorded data from MIT IS&T's server at 10 minute intervals and feeding the information into the web for visualization, shows the following patterns around campus below. Take note of the peaks and troughs in the graph below each map indicating the level of activity and community presence. High levels of connected devices glow as red circles while gray circles show areas with low traffic (Figure 3.14):

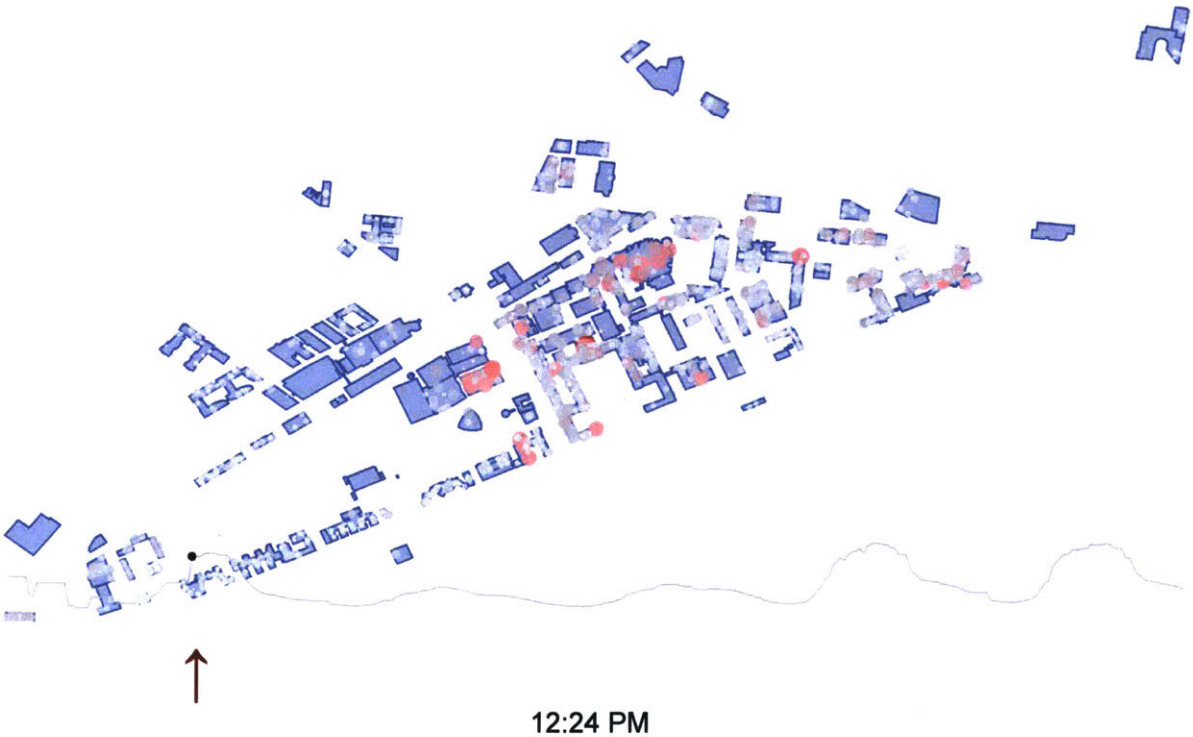
8:14 AM



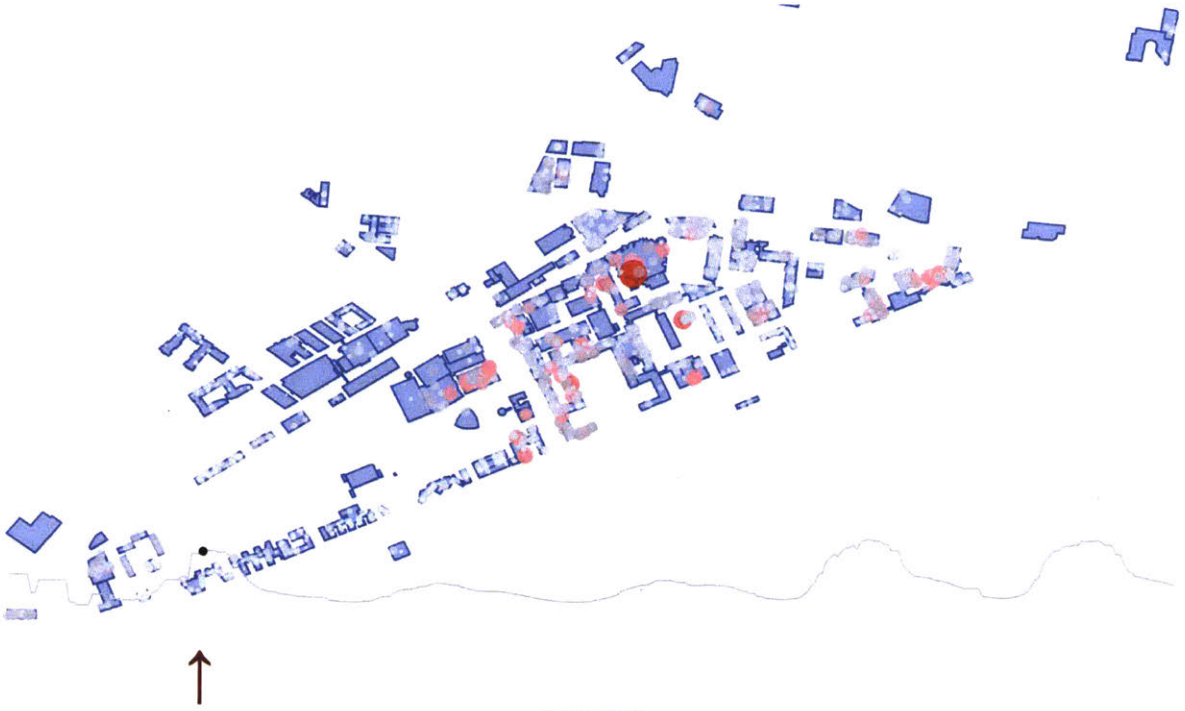
9:04 AM

10:19 AM

11:09 AM



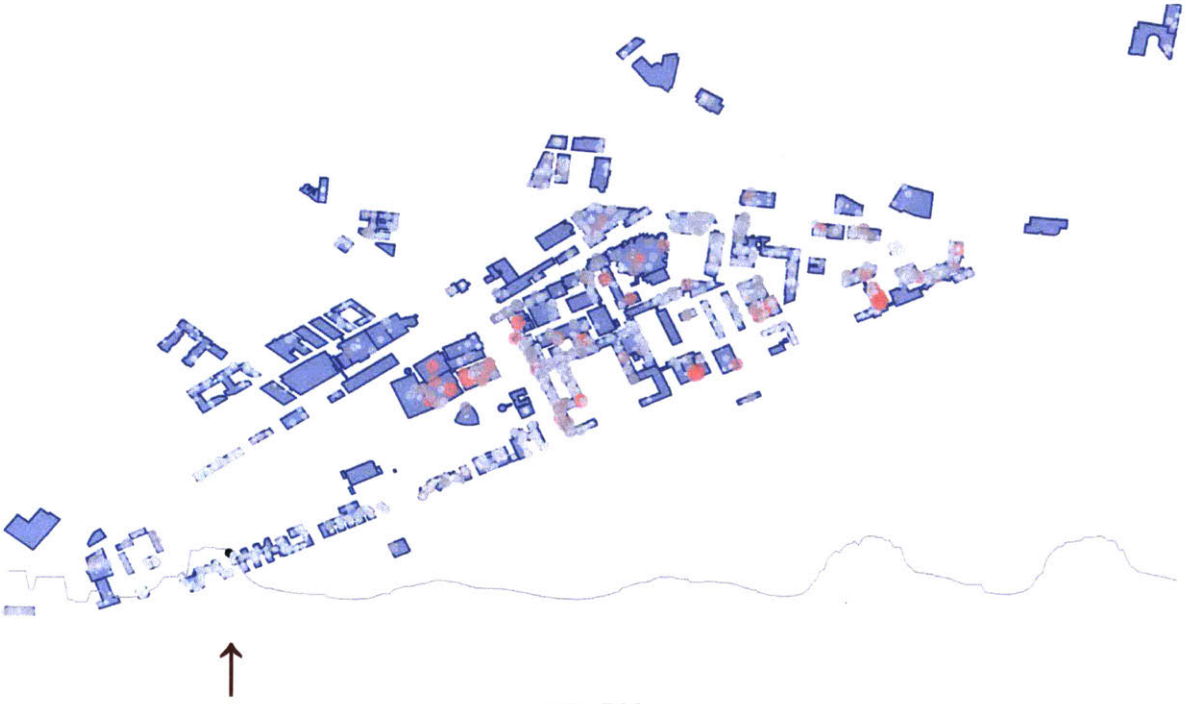
12:24 PM



1:39 PM

2:54 PM

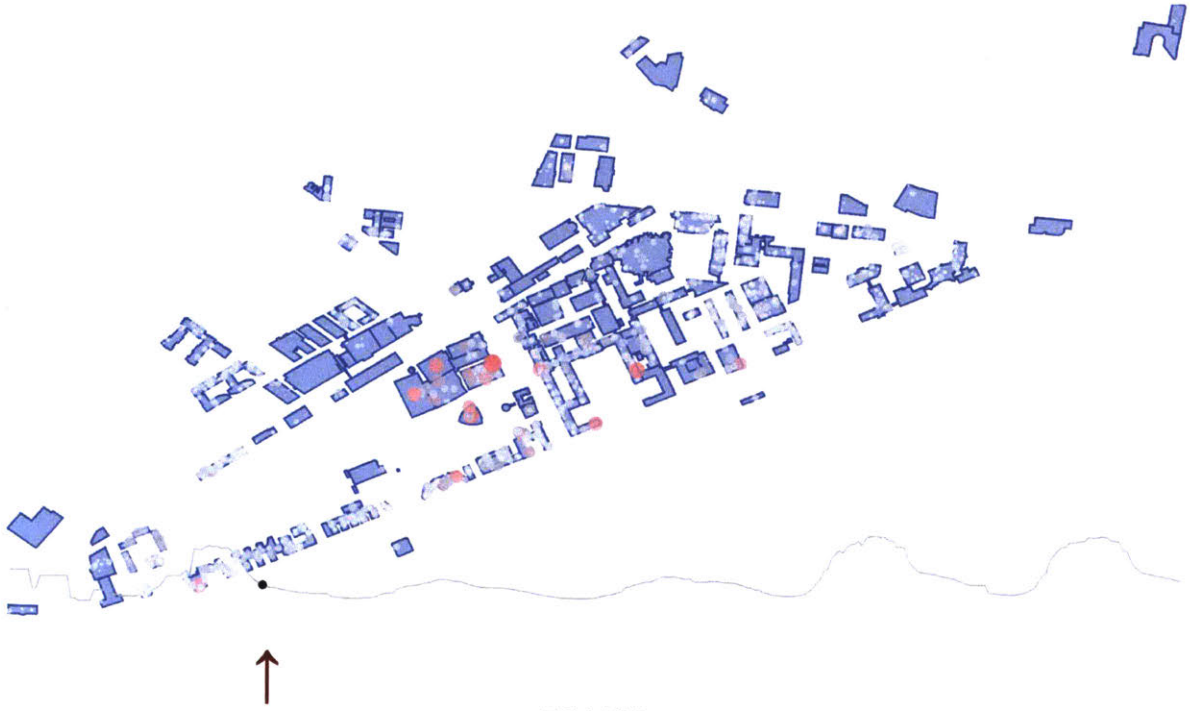
3:44 PM



4:34 PM

5:49 PM

6:39 PM



7:54 PM

8:44 PM

9:34 PM

10:49 PM

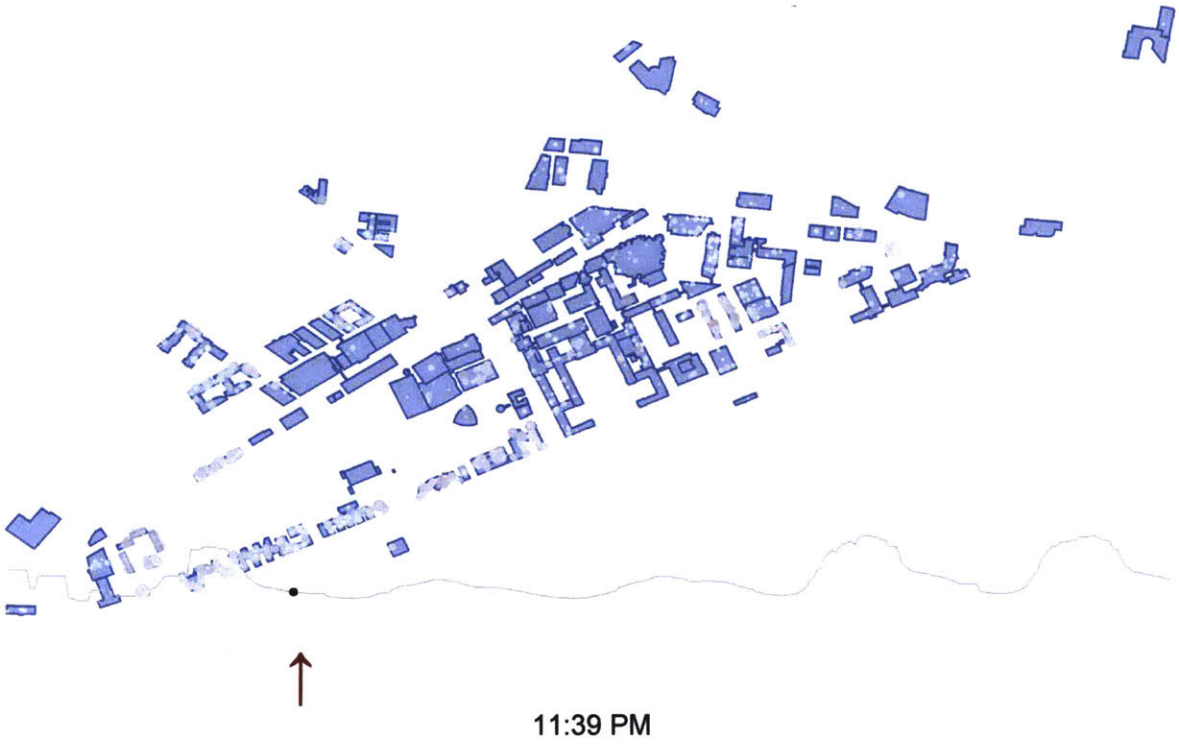


Figure 3.14 The number of connected devices to a WiFi access point throughout several days at MIT.

Overall, the MIT community begins to go online creating volumes of high traffic around 9am on the far eastern side of campus where the Sloan School of Business is located. Pockets of high traffic start to appear in just one hour around 10am around the middle of campus near Stata Center and towards the Stratton Student Center. The campus bustles with even more traffic at noon with highly concentrated spots at Stata Center up until around 3pm, where the WiFi activity is not nearly as high in concentrations but still fairly crowded between 4pm and 8pm. At 8pm, the far eastern side of campus loses its mid-high traffic and has fewer connected devices. Around 9pm, the majority of the MIT community migrates toward the west side of campus in low concentrations while parts of mid and west campus continue on with activity, and later wanes with the exception of the MIT's Muddy Charles Pub. Approaching midnight, mid-high WiFi traffic disappears completely leaving only low concentrations of people connected to online activity throughout the night only to repeat the cycle again beginning at 8am.

In Figure 3.15, the plot shows a correlation matrix between the number of connected devices and geographic location in latitude and longitude. Creating the plot seeks to answer the

following question, is there a correlation, or statistical relationship, between the number of connected devices and geographic location? In other words, do the number of devices and their owners appear in certain locations because of better connectivity more than other access points around campus? According to the results, the plot shows very little correlation (see Appendix C for the estimation of correlation) between time and latitude and longitude, 0.08, to be exact. A result of 1 (dark blue) indicates a strong correlation between variables and 0 (white) otherwise. As the data suggests, people do not move around campus to seek better WiFi connection. As expected however, there exists a strong dependence between latitude and longitude variables as well as itself.

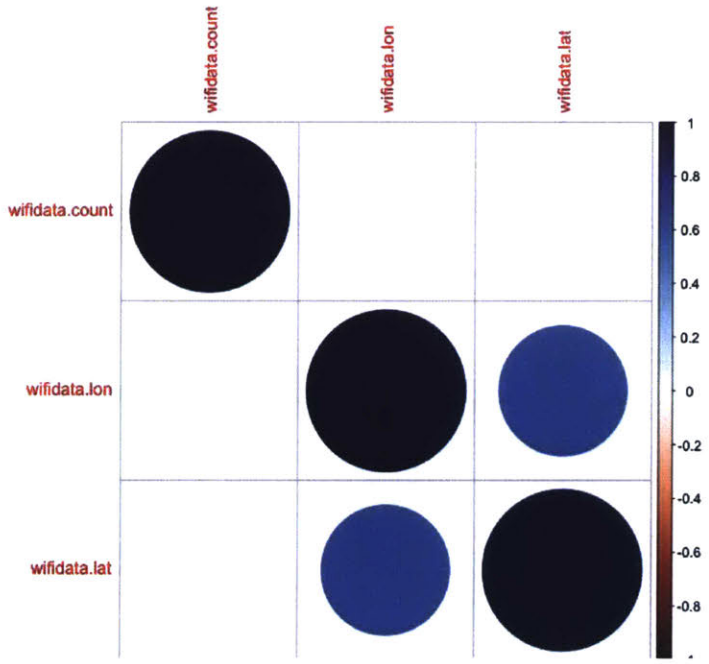


Figure 3.15 Cross-correlation between longitude and latitude with the number of connected devices.

3.3.1 Stata Center's WiFi Activity

Zooming into the Stata Center building, what does the crowd activity inside of Stata look like? When can facility managers or designers anticipate crowds to happen? Is there a dependent relationship between the time of day (and/or day of the week) and the number of devices connected at Stata Center? This section answers such questions in which the dataset ranges

from September 2014 to April 2015, offering a glimpse into crowd activity, or anonymous logging of devices recorded every 25 minutes by every WiFi Access Point around campus. At this point, the campus WiFi dataset has been trimmed to focus on Stata Center in this section and its values do not differentiate whether a device is a stationary computer sitting on a desk or a mobile device belonging to someone, but as the analysis in this section shows, the activity of individuals connecting to MIT's WiFi network fluctuates over time.

An overall picture of WiFi access points within Stata Center itself can be seen in Figure 3.16a and Figure 3.16b where connected device activity occur.

Figure 3.16 (a)

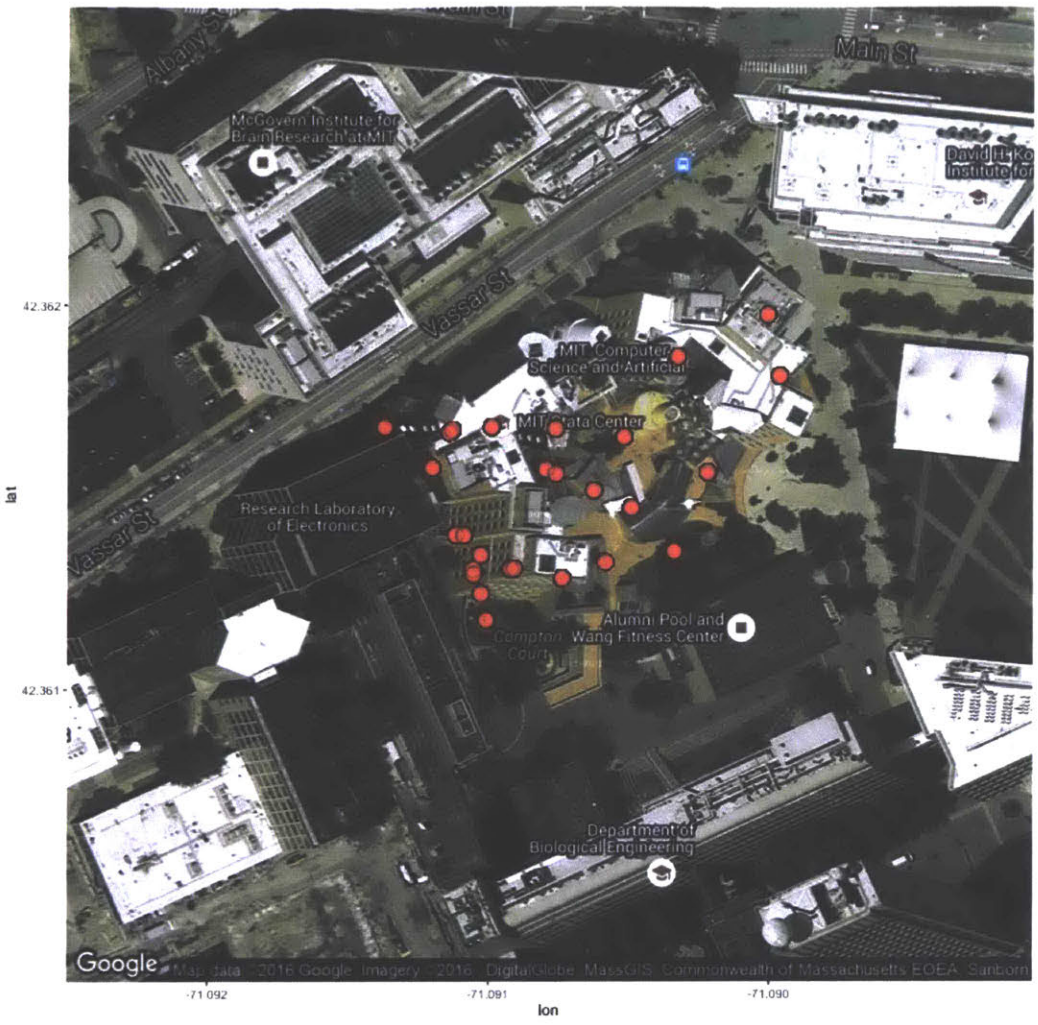
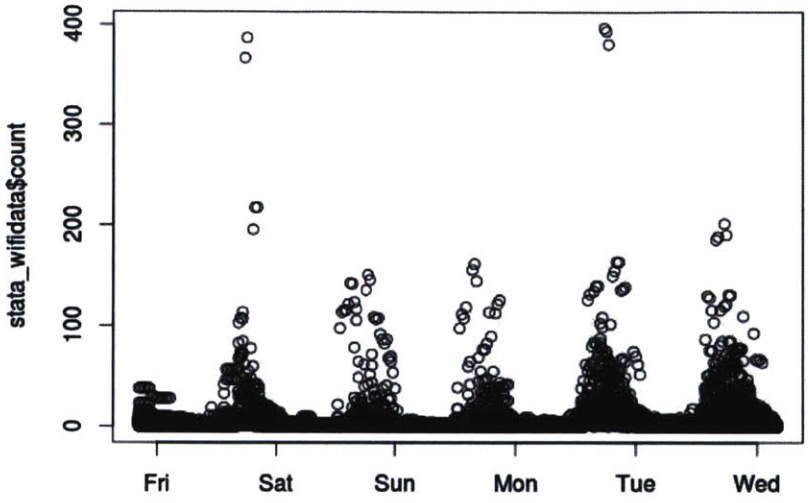
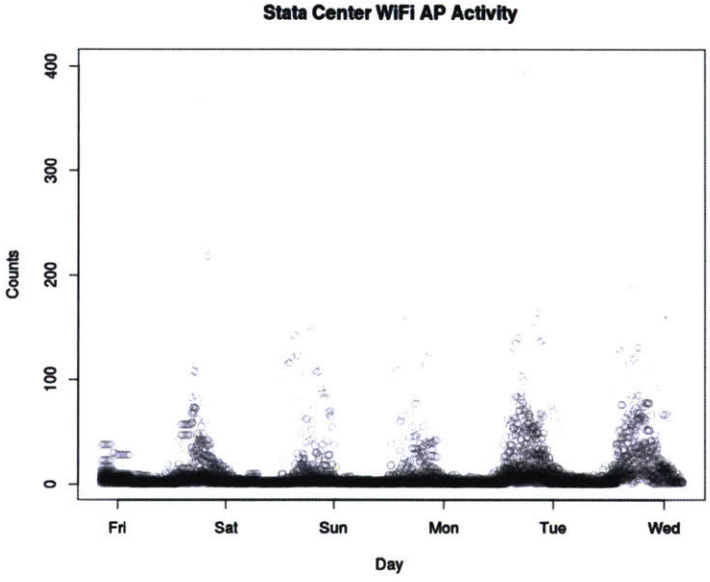


Figure 3.16 (b)

The dataset in this section is from September to October 2014. At the beginning of the school year, peak activity occurs with a minimum average of nine people connected at any time of the day. In the top diagram of Figure 3.17(a), there are some outliers in the data -- a total count reaching up to 400 connected devices on Friday and Monday in September. The outlier most likely suggests that a major event occurred in Stata Center around that time, which was on September 26 and 29 around 5pm and 7pm respectively. To clarify the distribution of points, the bottom plot in Figure 3.17(b) shows darker or dense points at the bottom of the graph and tapers to a lighter and tight distribution of points towards a device count of 400. Darker points show frequent occurrences while lighter point show less frequent occurrences.



(a)



(b)

Figure 3.17 (a) A full plot of the days and number of connected devices throughout Stata. (b) A clarified plot showing the density or more frequent activity occurring in Stata, and lighter or less activity occurring in large numbers at Stata Center.

In other words, the lighter colored points in the range between the 200 to 400 device count on Friday and Monday are not typical occurrences over at Stata Center, but the range between 1 to 50 connected devices on almost every day of the week is more common to occur.

What does Stata look like hourly? Figure 3.18 shows that the distribution of points are left skewed, or that the tail end of the data takes up the left side of the graph while the majority of the data occupies the right part of the diagram. The density of connected devices appear nearly constant up until 11am and then gradually ascends to a peak between 4pm - 6pm and then descends until 11pm.

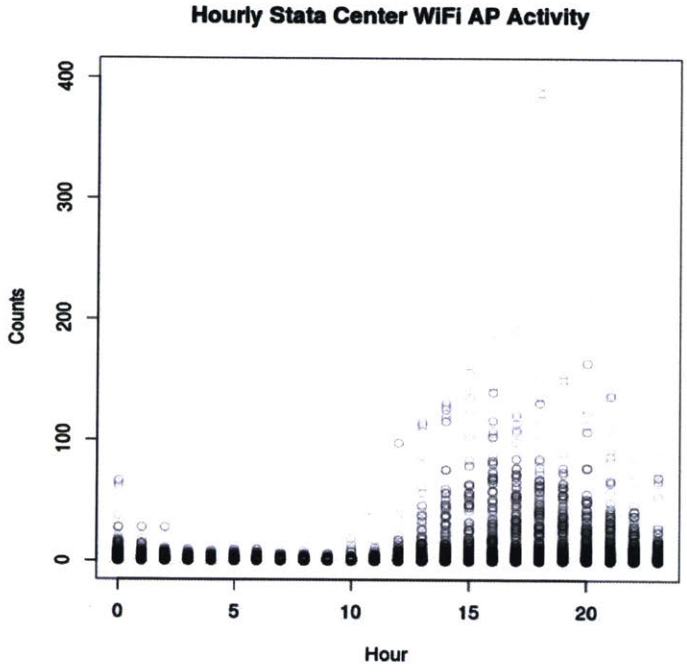


Figure 3.18 The WiFi data at Stata Center shows a left skew.

If we remove the outliers and just look at the distribution up to a device count of 150 (Figure 3.19, left), the data can now be seen as perhaps a more representative distribution of WiFi activity. Zooming in further for clarity up to a device count of 40 (Figure 3.19, right), the data points become more apparent as seen in Figure 3.20 -- this cross section of data points confirm that a lower number of devices occur at all hours while a high frequency of connected devices do not cluster at all hours.

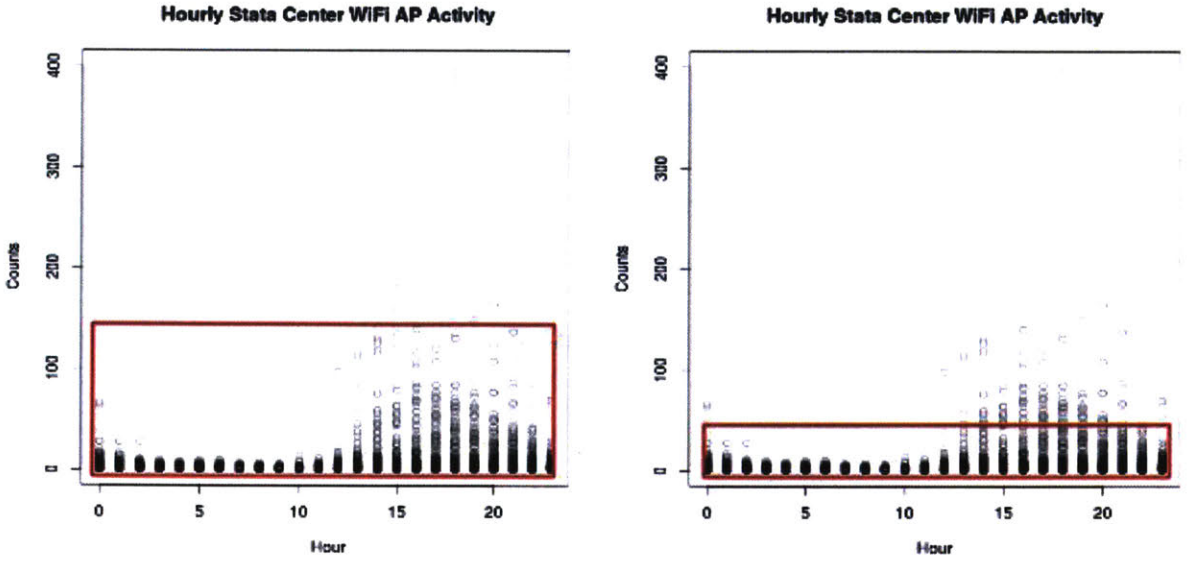


Figure 3.19 (left) Removing outliers from the distribution. (right) Zooming in further at 40 device counts.

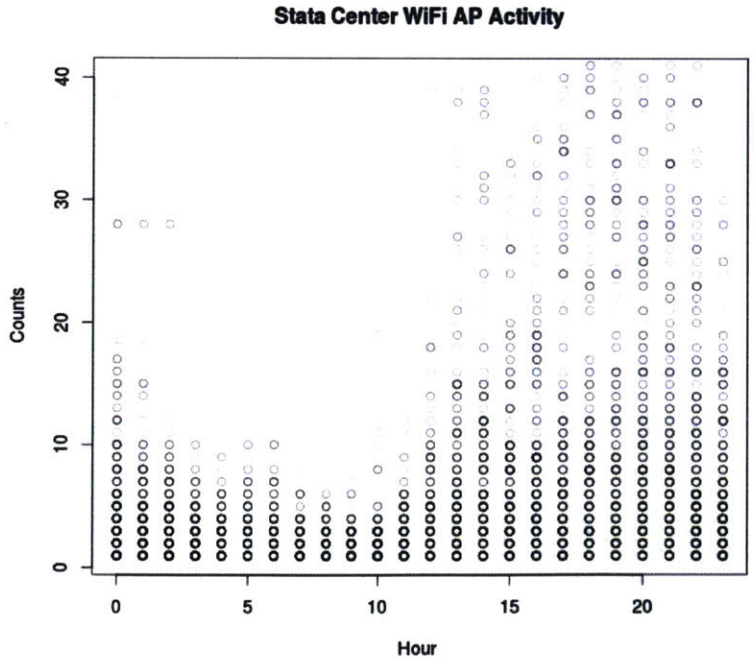


Figure 3.20 The higher the recorded device count is, the less it occurs. The darker the points, the more frequent the number of devices occur at a particular hour.

As in the previous cluster analysis of Stata Center using image data points, looking for crowd clusters via device activity is also evident in the dataset. Figure 3.21 shows the number of connected devices relative to the WiFi access point locations, the plot also shows that two clusters form the majority of device activity indoors. Breaking down the data even further into four segments as seen in Figure 3.22, there are four micro clusters of device activities happening within Stata Center.

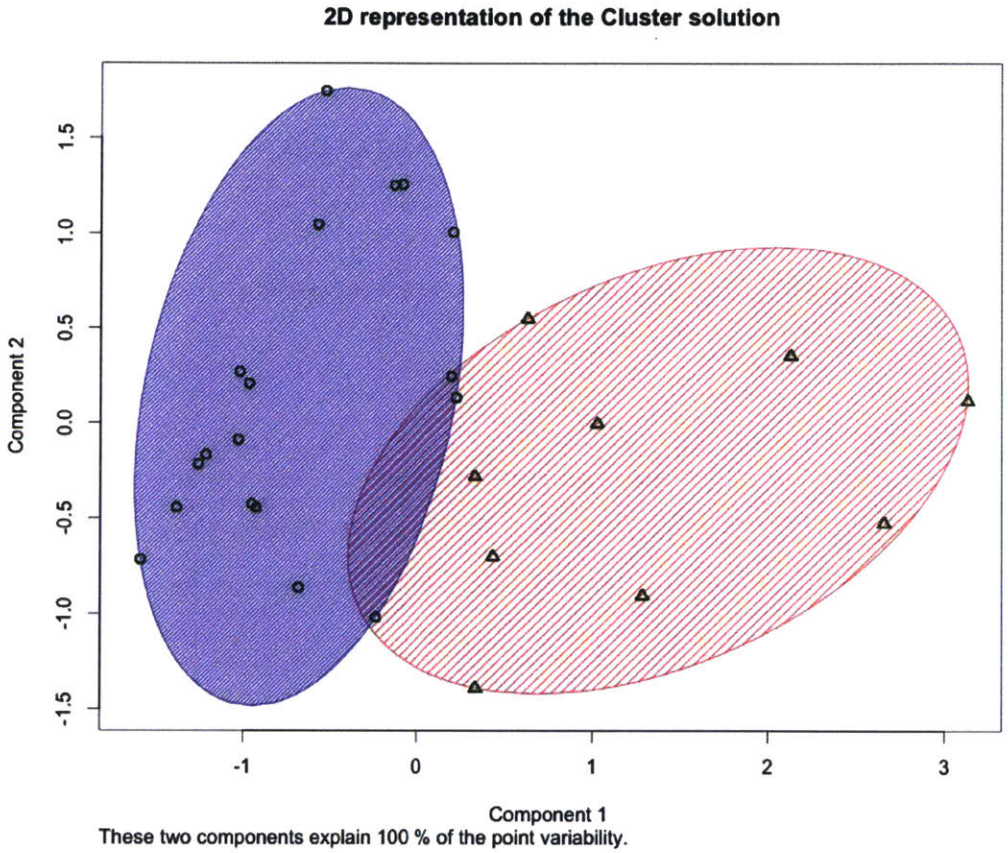


Figure 3.21 WiFi device activity and two major clusters found.

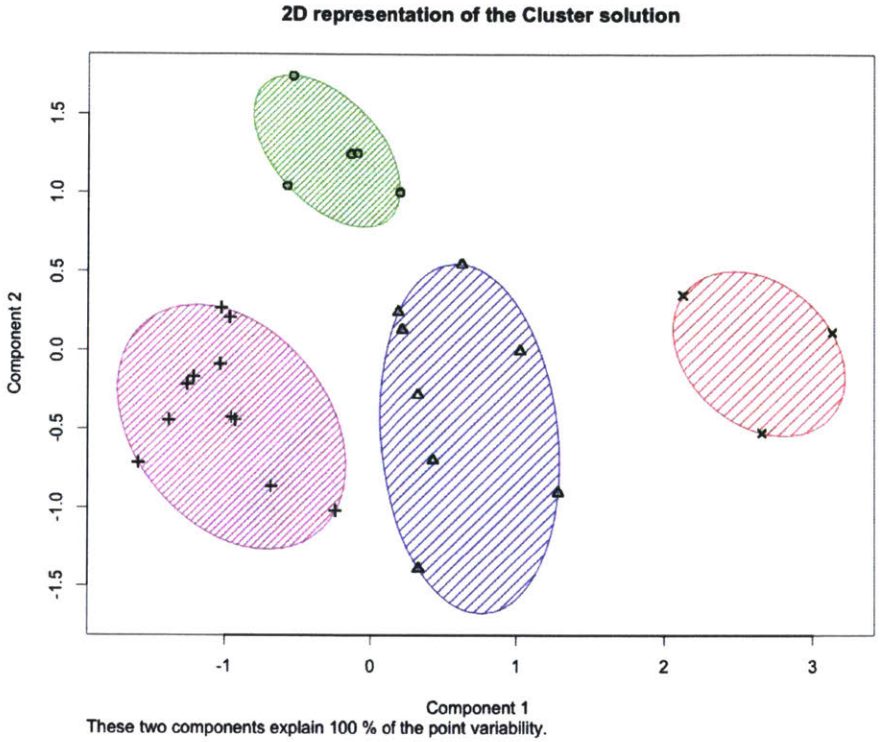


Figure 3.22 Breaking down the data into smaller segments show that four clusters actually exist.

Plotting the WiFi data points and the cluster solution shows that individuals and their devices connect to four major indoor locations. Factoring time into the k-means cluster, the optimal segments are k = 5 (Figure 3.23). Figure 3.24 demonstrates a shift in device connectivity as a function of time in the cluster solution.

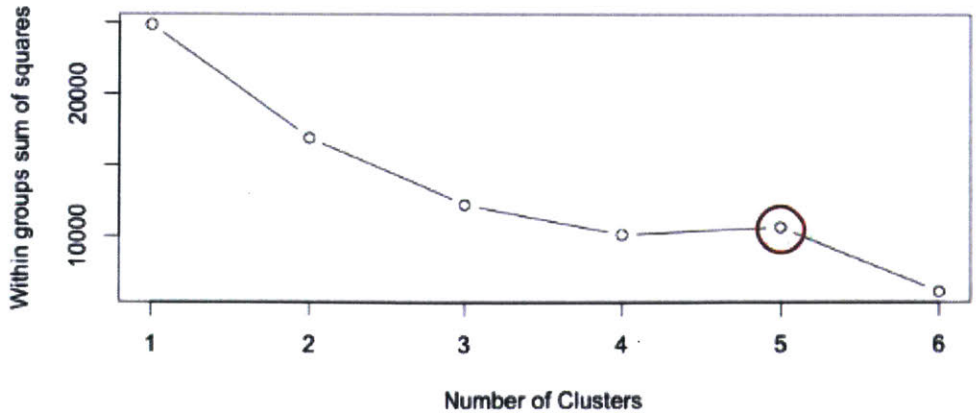


Figure 3.23 Determining the optimal k to segment the data. Using the "elbow method," k=5.

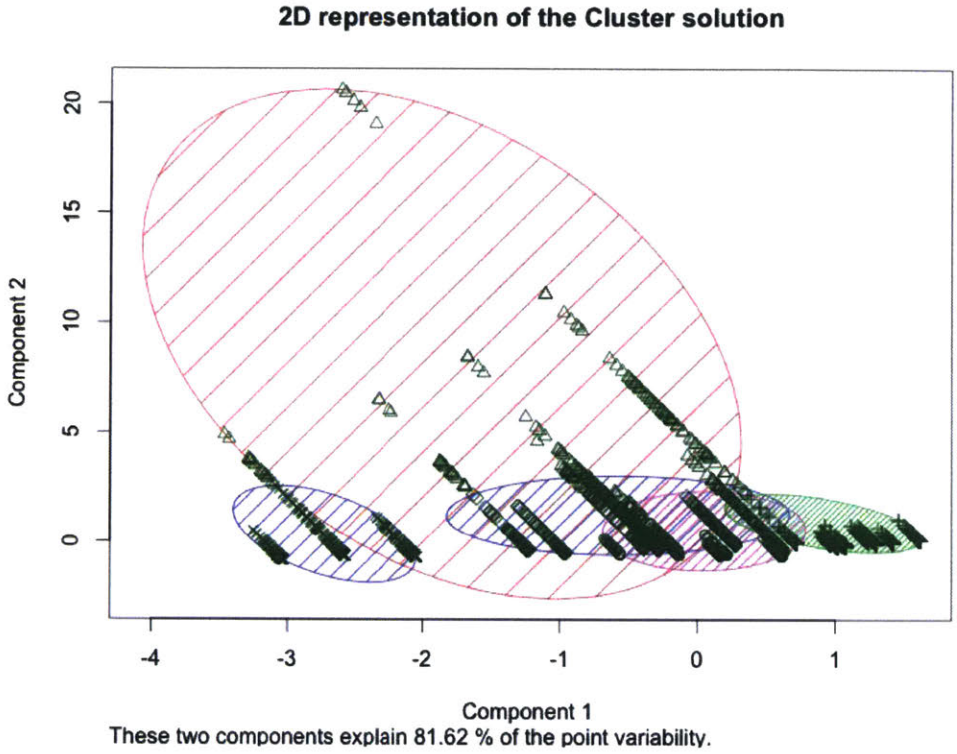


Figure 3.24 Cluster solution with time as a variable.

3.3.2 Predicting Crowds

This research relies on several data points that involve time as a variable, or a collection of data recorded over a period of time. Variables recorded over time include the frequency of anonymously connected users to WiFi access points and the number of images captured by individuals about buildings. The usefulness in time series information allow us to not only observe the crowd trends that occur within the interior and exterior of a building, but also serves as a practical tool for designers and building management to forecast crowd or even utility patterns into the future. In 2006 for example, Singapore Airlines placed an order for twenty Boeing 787-9s and twenty-nine new Airbus planes, twenty A350s, and nine A380s (super jumbo jets).[12] The airline's decision to expand its fleet "relied on a combination of time series analysis

[12] Wardel, Jane. "Singapore Airlines Orders Airbus Planes". The Washington Post. July 21,2006 http://www.washingtonpost.com/wp-dyn/content/article/2006/07/21/AR2006072100744.html

of airline passenger trends and corporate plans for maintaining or increasing its market share[13]." With the information crowd-based information at hand, how can building managers use this research application's specific time series model to determine short-term forecasts of when crowds might form in and around buildings? How can the information inform designers where to place objects or design new public spaces? How might an architectural historian use the information to better understand how people utilize public spaces versus the architect's intent?

This chapter discusses the statistical methods and algorithms used for analyzing and generating short-term forecasts based on the trend patterns from the dataset.

This section will explore two predictive modeling algorithms: Linear Regression and Holt Winters for time series data. In statistics, the linear regression models the relationship between variables and is extensively used in practical applications. In this case, linear regression is used to fit a predictive model to the observed data set -- given a variable *y* and a number of other variables $X_1, \ldots X_n$, we want to predict the likelihood of future crowd formation at all hours at Stata Center, a linear regression model shows an estimate by fitting a line through most of the points in Figure 3.25.

[13] Cowpertwait, S.P. Paul and Andrew Metcalfe. Introductory Time Series with R. Springer: New York. 2009.

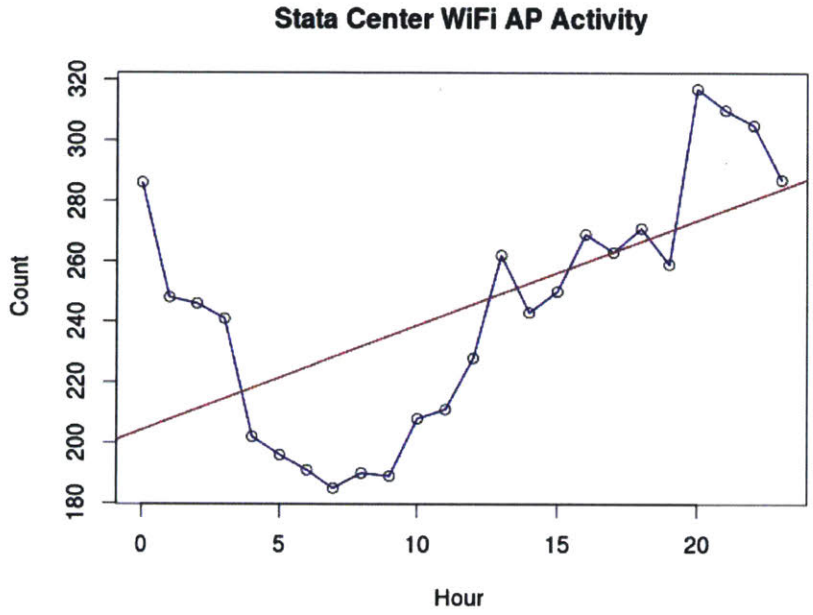


Figure 3.25 A linear regression through the data points.

The line is described in the equation as $Y = 1.53233 + 0.56977X$, a right upward slope indicating a positive correlation between two variables, count and hour. In other words, there is a relationship between the variables where the count variable is affected by the hour variable and vice versa. The line, however, does not quite fit the data points, so the Holt Winters method for short term time series forecasting delivers another approach for analysis.

Figure 3.26 shows a time series of device activity for Stata Center only, Saturday and Tuesday reached nearly 400 device counts and is the largest amount than any other time during the week. Figure 3.27 pinpoints the location of the sudden spike in device counts to room 32-123, a class lecture hall/auditorium and displays a clean plot of the data. To make short-term forecasts of the time series data, applying exponential smoothing assigns more weight to recent observations and less weight to older observations, seen in Figure 3.28. The exponential smoothing method contrasts with other time series techniques such as Single Moving Averages where the past observations are weighted equally. The short term prediction segment (Figure 3.29) yields an accuracy based on an 80%-95% confidence interval that determines whether the device population will peak above or below 400 connected devices.

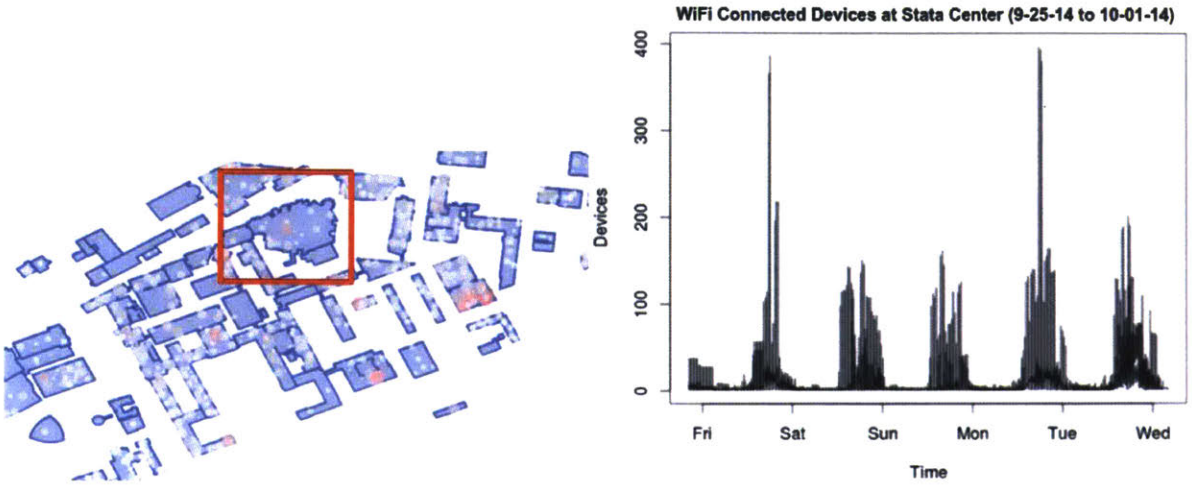


Figure 3.26 Extracting Stata Center's device count activity.

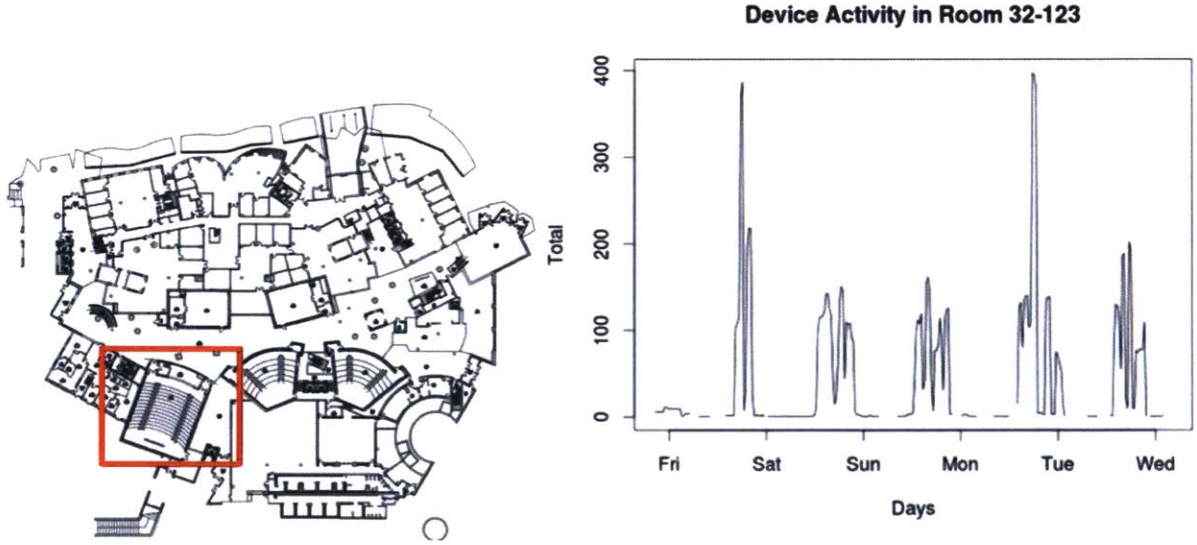


Figure 3.27 Extracting device activity for Room 32-123 only.

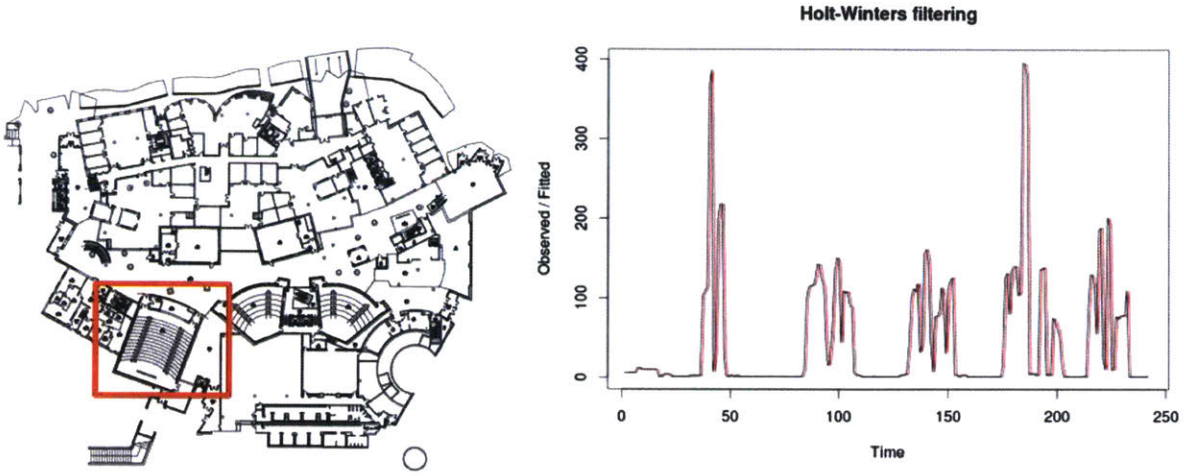


Figure 3.28 Exponential smoothing of the time series data to fit a predictive model.

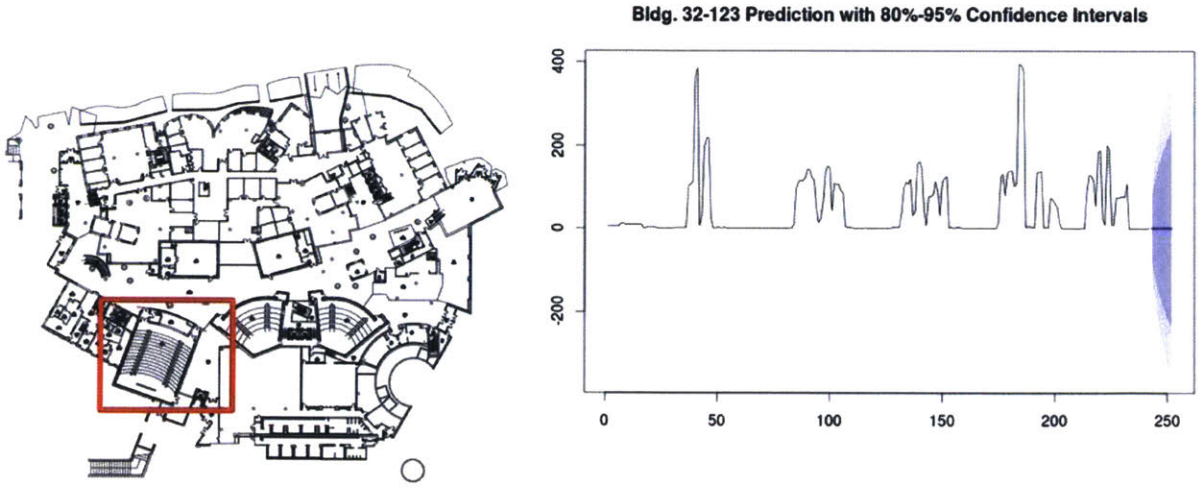


Figure 3.29 The forecast interval, with 80% confidence that the population will peak near 400 connected devices, and a 95% confidence interval that it will be closer to 200 devices.

3.4 Stata Center through Text

Gathering the keywords and captions from the metadata within each image reveals the word choices that people select to describe Stata Center. For historical purposes, this information can prove useful when learning how the generations of individuals perceive built works over time. One example of debate and controversy over an architectural/engineering construct was the Eiffel Tower in Paris, France by Gustave Eiffel. In 1887, the work received hurling insults and protest from artists and the general public. The Eiffel Tower has been labeled as "this truly tragic street lamp," or "this mast of iron gymnasium apparatus, incomplete, confused and deformed," or more thoroughly:

> We come, we writers, painters, sculptors, architects, lovers of the beauty of Paris which was until now intact, to protest with all our strength and all our indignation, in the name of the underestimated taste of the French, in the name of French art and history under threat, against the erection in the very heart of our capital, of the useless and monstrous Eiffel Tower which popular ill-feeling, so often an arbiter of good sense and justice, has already christened the Tower of Babel...Is the City of Paris any longer to associate itself with the baroque and mercantile fancies of a builder of machines, thereby making itself irreparably ugly and bringing dishonour[14]?

The cultural tastes of the era would evolve into an amourous adoration for the Eiffel Tower as the city's central icon for generations to appreciate. With that in mind, what words do individuals associate with Stata Center in the current context of architectural styles and utility? How does the general public react to a building whose forms are non-traditional in design and construction in words?

Figure 3.30 shows the choice of words and frequency of those words in the image dataset. Not surprisingly, "stata center," "mit," and "frankgehri" appear as the top three tagged words. Looking further into the terms, Figure 3.31 shows the rest of the vocabulary used for describing the building. Ignoring the the top three terms and directing one's attention to the terms in smaller font, other words that people have used to describe Stata Center include "childlike,"

[14] La Tour Eiffel. Official website. http://www.toureiffel.paris/en/everything-about-the-tower/themed-files/71.html. Accessed June 9, 2016.

"deconstructionist," "whimsical," "fav," "surreal," "gehryeffectbuilding," "makesyouwonder," and "hmmm" to name a few.

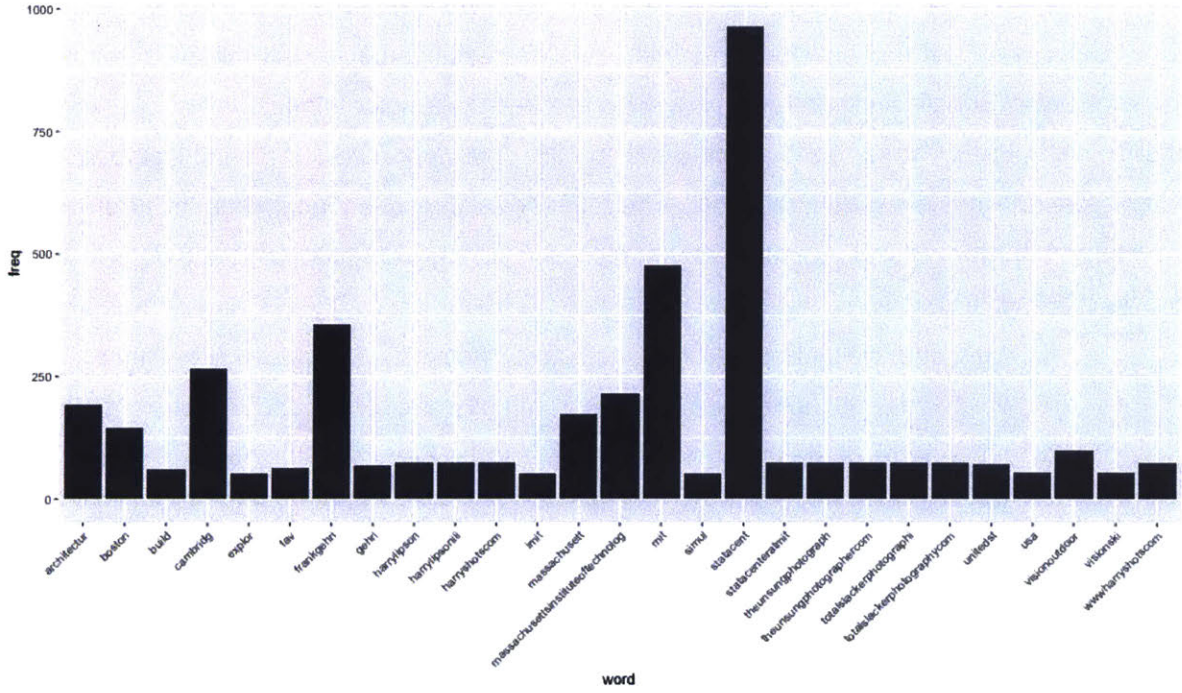


Figure 3.30 Stata Center word frequency.

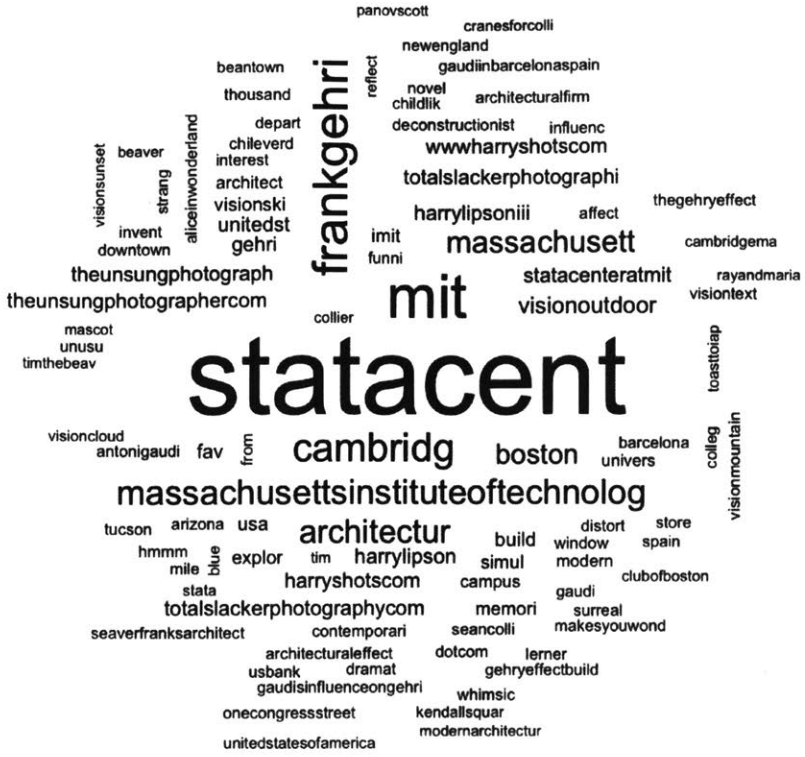


Figure 3.31 100 most frequent terms associated with Stata Center.

The smaller font words become of interest precisely because of its descriptive nature and public reaction to Stata Center itself. Overall, the general impression from the public leans toward amusement and/or ponderance. Taking the term association of "hmmm," for example, shows a correlation between "hmmm" and other words. Figure 3.32 shows the correlation coefficient under each term associated with the target query word. A coefficient close to 1 indicates a strong positive linear relationship between terms, a scatter plot would show a positive linear slope. Figure 3.32a and Figure 3.32b displays an organized plot of correlation rank and terms.

```
$hmmm
                affect          aliceinwonderland                antonigaudi     architecturaleffect
                  1.00                       1.00                       1.00                    1.00
 architecturalfirm                    arizona                  barcelona                    blue
                  1.00                       1.00                       1.00                    1.00
          childlik                    chileverd                     depart                  dotcom
                  1.00                       1.00                       1.00                    1.00
          downtown                     dramat                     explor                   funni
                  1.00                       1.00                       1.00                    1.00
             gaudi  gaudiinbarcelonaspain gaudisinfluenceongehri       gehryeffectbuild
                  1.00                       1.00                       1.00                    1.00
             imit                    influenc                   interest                  invent
                  1.00                       1.00                       1.00                    1.00
           lerner                  makesyouwond                       mile                   novel
                  1.00                       1.00                       1.00                    1.00
 onecongressstreet  seaverfranksarchitect                       simul                   spain
                  1.00                       1.00                       1.00                    1.00
            store                      strang                    surreal        thegehryeffect
                  1.00                       1.00                       1.00                    1.00
         thousand                      tucson                      unusu                  usbank
                  1.00                       1.00                       1.00                    1.00
          whimsic                  contemporari                    distort                    from
                  1.00                       0.98                       0.98                    0.98
deconstructionist                      modern                  architect
                  0.96                       0.87                       0.79
```

Figure 3.32 (a) List of words correlated with the target keyword.

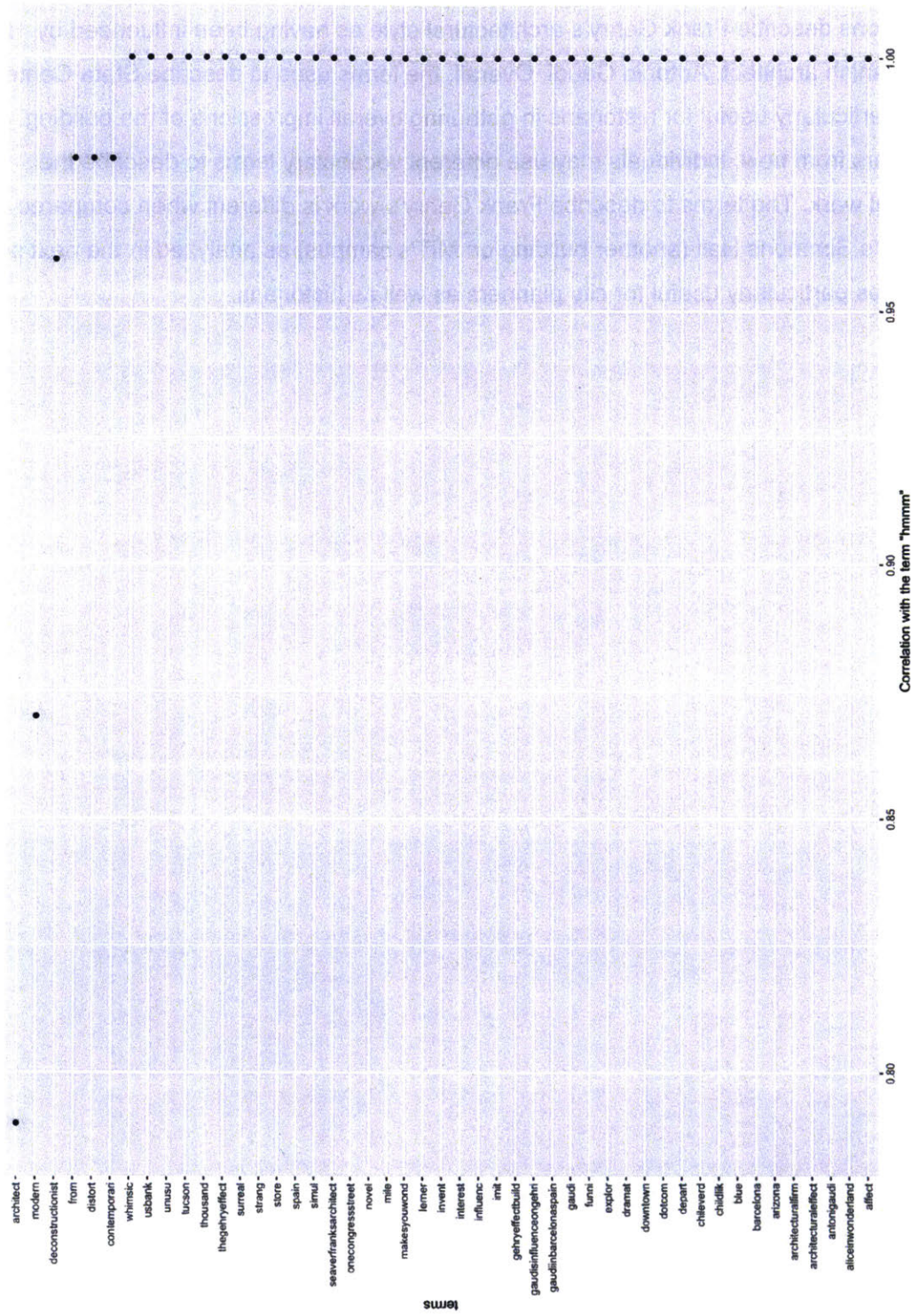


Figure 3.32 (b) Correlation plot.

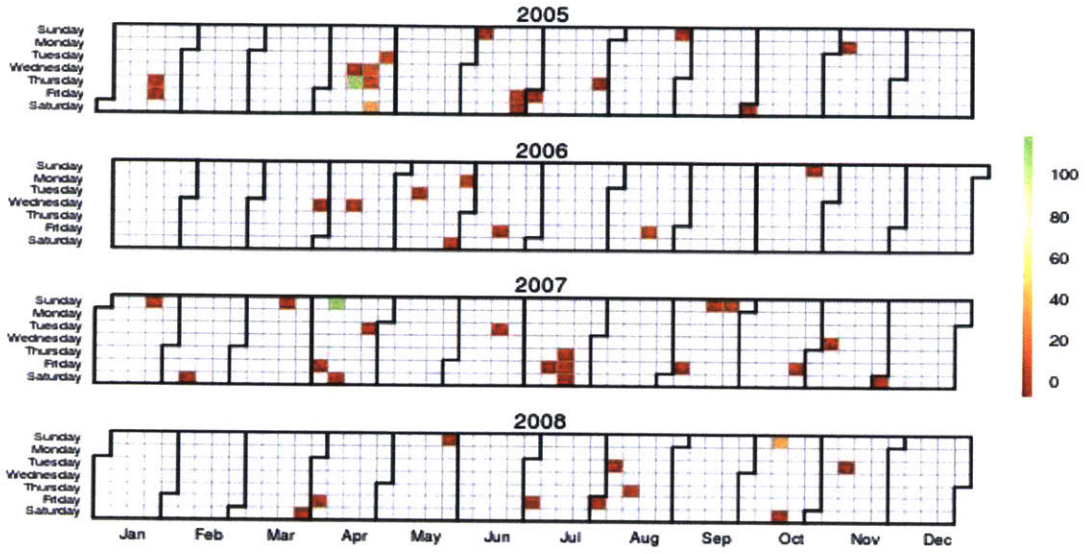
Some captions describe Frank Gehry's architectural style as having been influenced by 19th century Spanish architect, Antonio Gaudi. Overall, the terms used to describe Stata Center becomes particularly useful for historians in gathering overall impressions of the building. Several years from now, individuals may use different vocabulary terms to describe the architectural work. The terms to describe Frank Gehry's work is different when compared to Steven Holl's Simmons Hall (another building on MIT's campus) as analyzed in the next section, and becomes particularly useful for city planners as well as historians.

3.5 Analyzing Simmons Hall

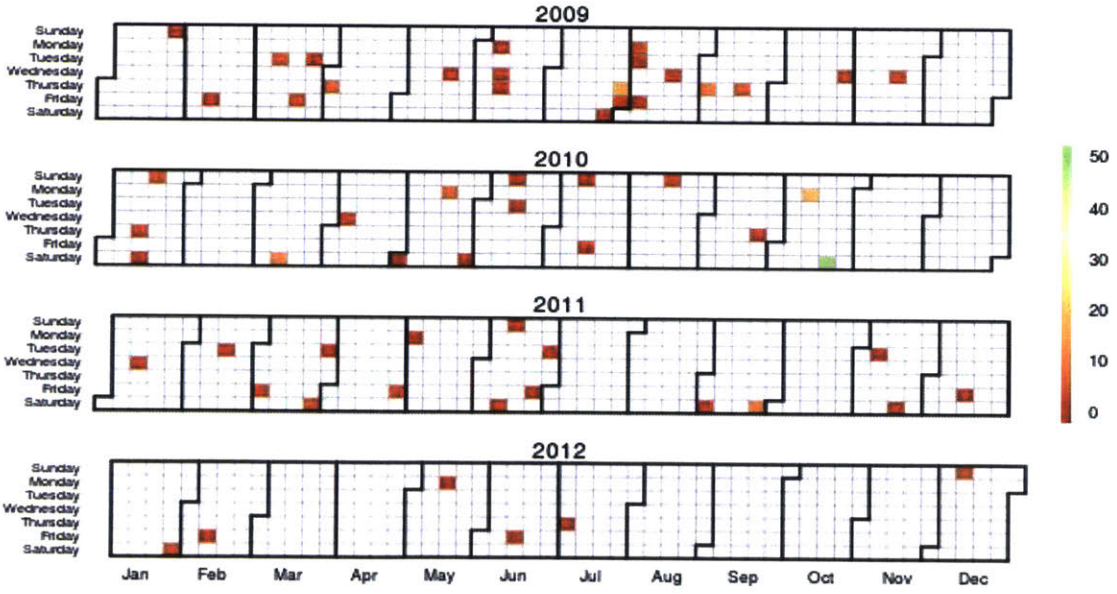
Figure 3.33 Simmons Hall Residence

Simmons Hall is an undergraduate dormitory on MIT's campus that was built between 1999 - 2002 (Figure 3.33). When compared to Stata Center, Simmons Hall had less image activity throughout the years of 2005 to 2015 (Figure 3.34). The data allows us to speculate the following questions: Are the low image counts due to Simmons' location? Does Stata Center receive more attention because of its proximity to the Boston T subway station and main campus entrance? What if location isn't a factor and people photograph Stata Center more because the building's design appeals more to the masses than Simmons?

Calendar Heat Map of Simmon's Hall Image Activity, 2005-2008



Calendar Heat Map of Simmon's Hall Image Activity, 2009-2012



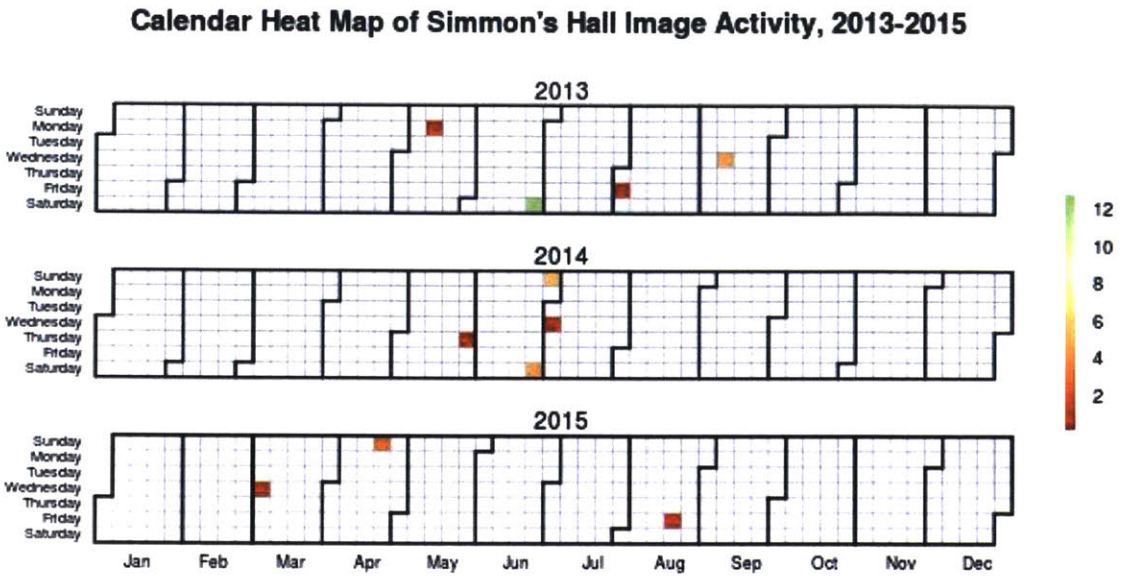


Figure 3.34 Simmon's Hall image activity

Figure 3.35a shows where the majority of individuals photograph Simmons Hall, following along the geometric length of the building along Vassar Street. A sample of the photos are seen in Figure 3.35b as well as a plot of camera positions since 2004 in Figure 3.35c.

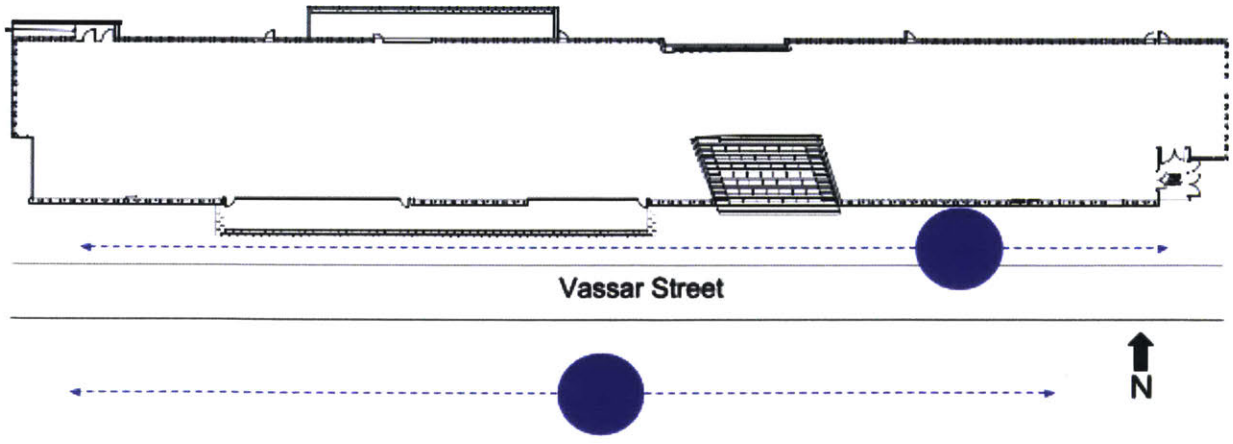


Figure 3.35 (a) Clusters along the street corridor.

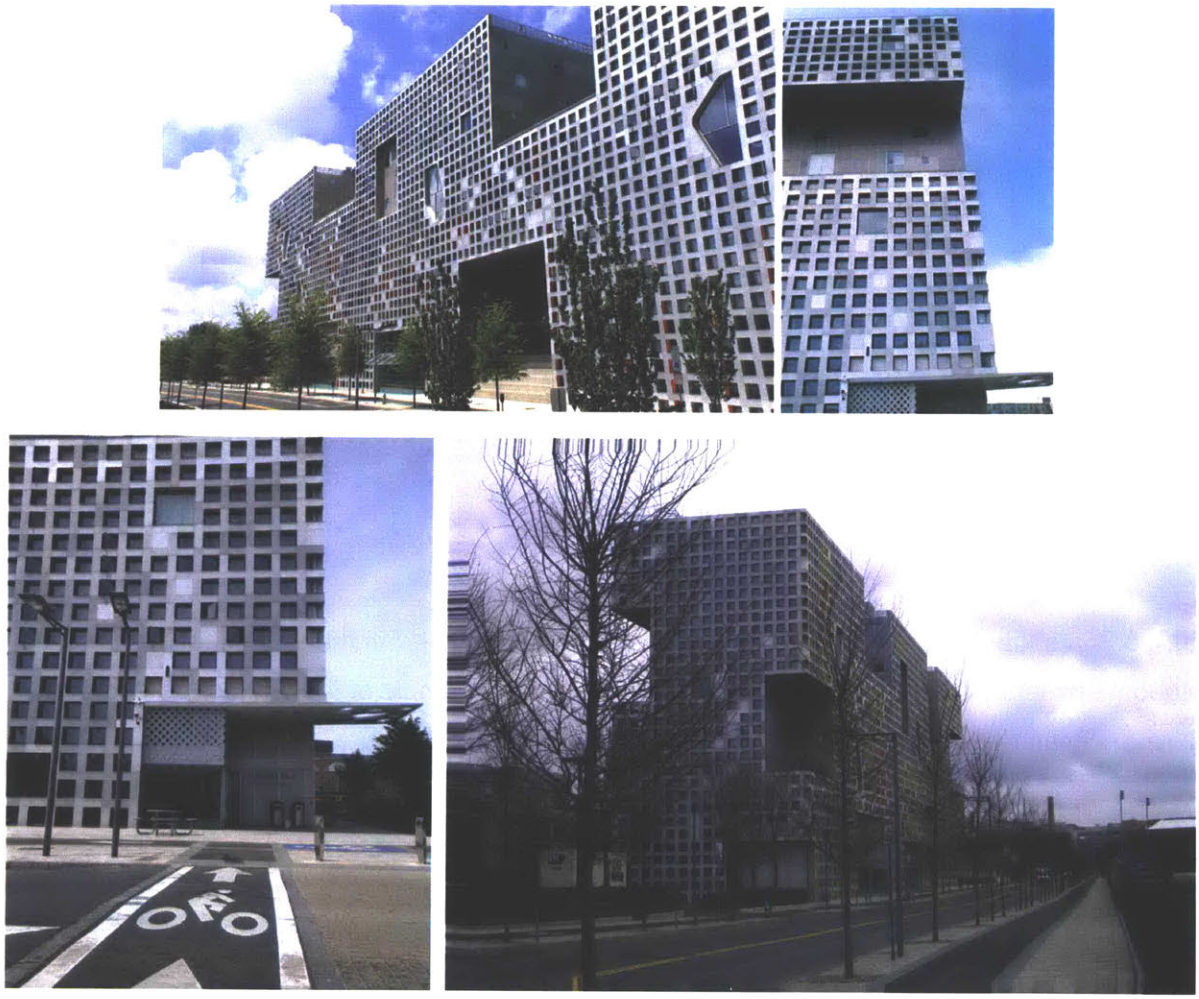


Figure 3.35 (b) A sample of photos taken and uploaded by the public.

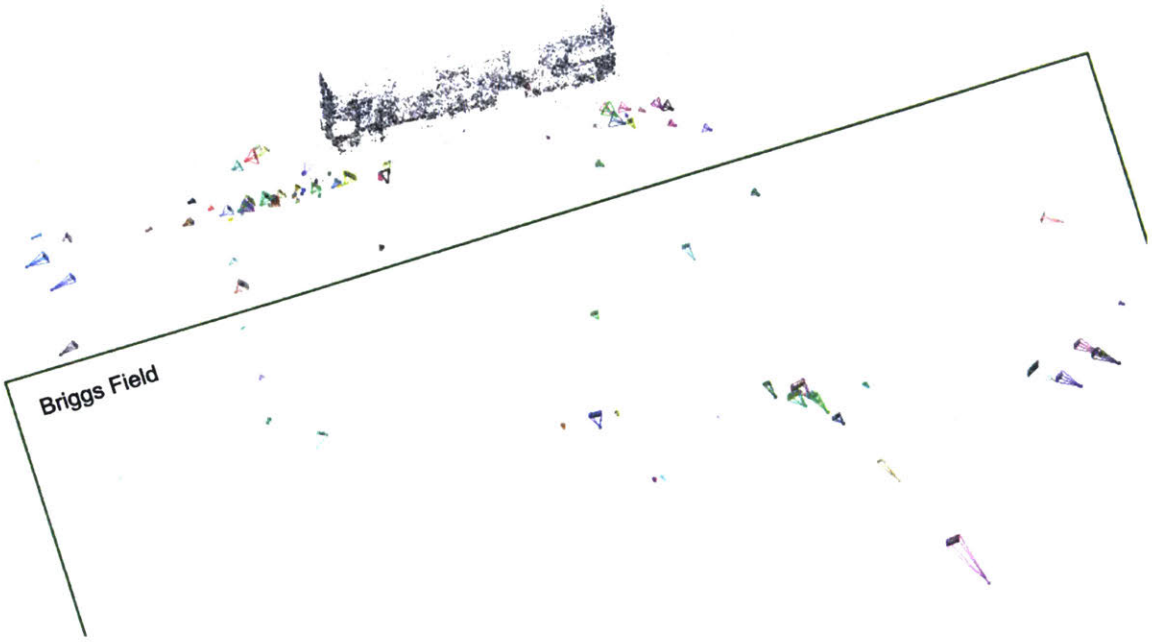


Figure 3.35 (c) Situated camera positions around Simmons Hall since 2004.

Simmons Hall began with low photo counts in the year 2004, but picked up in the year 2005 and 2007 with Spring time being the more popular season to photograph the building (Figure 3.36). The years 2008-2011 all show low photo counts throughout the year but sometimes display a spike of greater than 30 photographs taken around the Fall season (Figure 3.37). All in all, Simmons Hall does not receive many picture-taking visitors when compared to the amount that Stata Center receives.

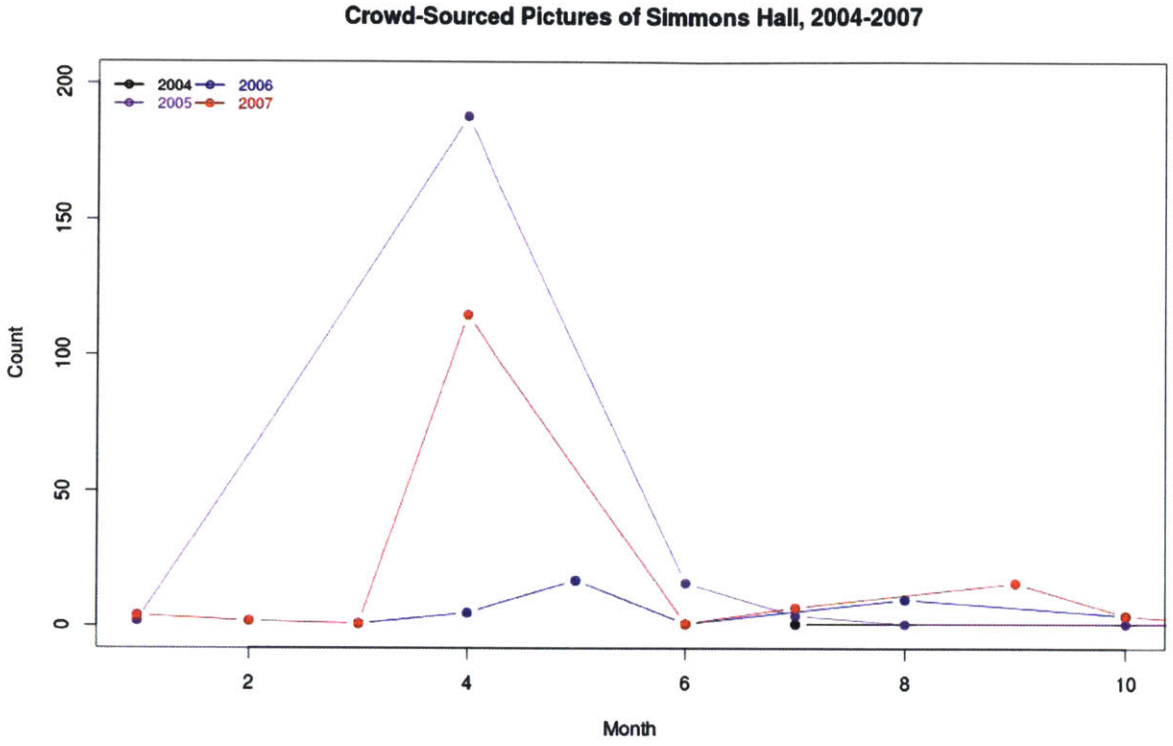


Figure 3.36 Number of photographs taken at Simmons Hall throughout the months of 2004-2007

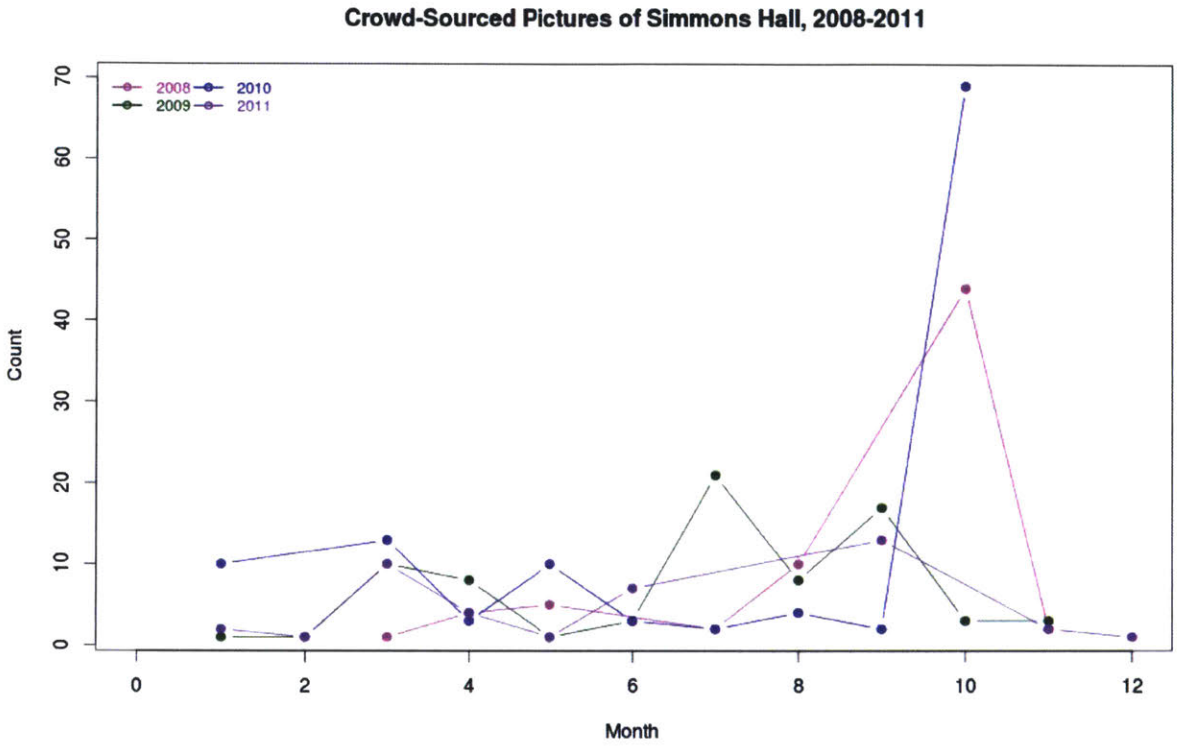

Crowd-Sourced Pictures of Simmons Hall, 2008-2011
2008
2009
2010
2011
Count
Month
0
10
20
30
40
50
60
70
0
2
4
6
8
10
12

Figure 3.37

3.5.1 Analyzing Simmons Hall Through Text

Amongst the many words associated with the set of Simmons Hall images, six keywords/captions arise as the most frequent. Table 3.4 demonstrates the top words and their frequency in the image dataset.

Word	Frequency
Boston	490
Steven Holl	547
Architecture	605
Cambridge	611
Fav	949
Simmons Hall	1464

Table 3.4 Word frequency for Simmons Hall.

Words such as "Boston", "Simmons Hall", or "Architecture" associated with the building does not reveal anything new, except one word that shows a preference for the built work -- "fav," or short for "favorite." The public sentiment of Simmons Hall is a positive one. Whether the building is a favorite amongst buildings on MIT's campus or a favorite amongst all buildings in Cambridge and Boston, or the world, is unclear. Figure 3.38 shows the word association or correlation between the word, "fav," and other terms. "Fav" is highly correlated with "mitmassachusetts" at 0.97 as well as "beantown" and "newengland" at 0.93. Architect Steven Holl's name is not as highly correlated when compared to the rest of the terms before "holl."

mitmassachusett	beantown	newengland
0.97	0.93	0.93
unitedstatesofamerica	unitedst	holl
0.92	0.89	0.88
usa	superfav	blackandwhit
0.83	0.79	0.76
blackwhit	blackwhitephoto	monochrom
0.76	0.76	0.76
massachusettsinstituteoftechnolog		
0.72		

Figure 3.38 Word associations and their correlation to the word "fav" found in the image captions.

Figure 3.39 (a and b) quantifies the number of terms of what people are saying about Simmons Hall in their photo captions.

Figure 3.39 (a) A word cloud plot listing the frequency of terms.

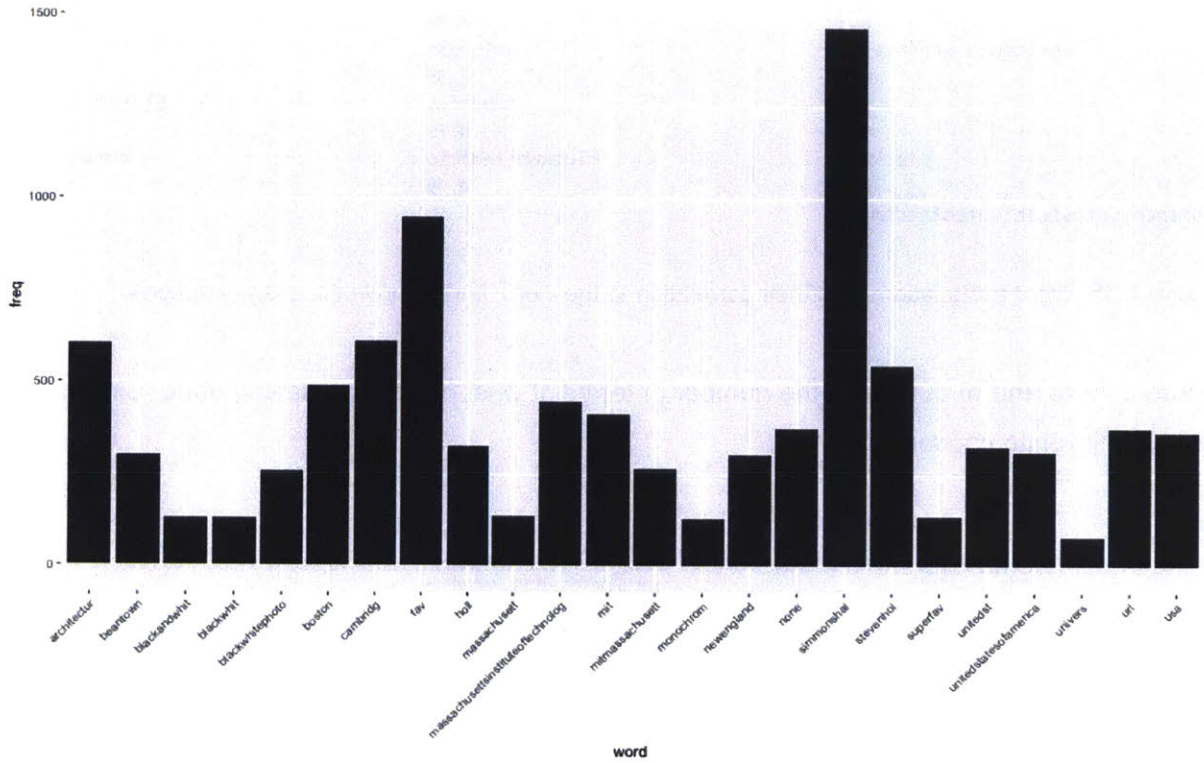


Figure 3.39 (b) Frequency term graph.

Based on the graph, the word "fav" is mentioned almost a thousand times in the image dataset, second to "Simmons Hall" reaching nearly 1,500 mentions in photo tags. For city planners, does this mean that more buildings in the aesthetic of Simmons Hall should be built in the city? Can designing more spaces by Steven Holl generate more revenue from travel and tourism? Overall, the text analysis helps planners, the general public, and historians understand what the current attitudes and reactions are towards Simmons Hall. Performing a text analysis on the same building several years from now might reveal a shift in attitude or remain the same.

3.5.2 Simmons through WiFi Activity

The Simmons Hall WiFi dataset contains 15,043 observations over a period of 4 days. The distribution of WiFi access points occur throughout the length of the building as seen in Figure 3.40. The average number of connected devices to each access point is 7 and the maximum is 62. The crowd activity remains evenly distributed throughout the day, but around 12PM and 7PM, the number of connected devices increase on the west side of the building -- the dining hall location (Figure 3.41 and Figure 3.42).

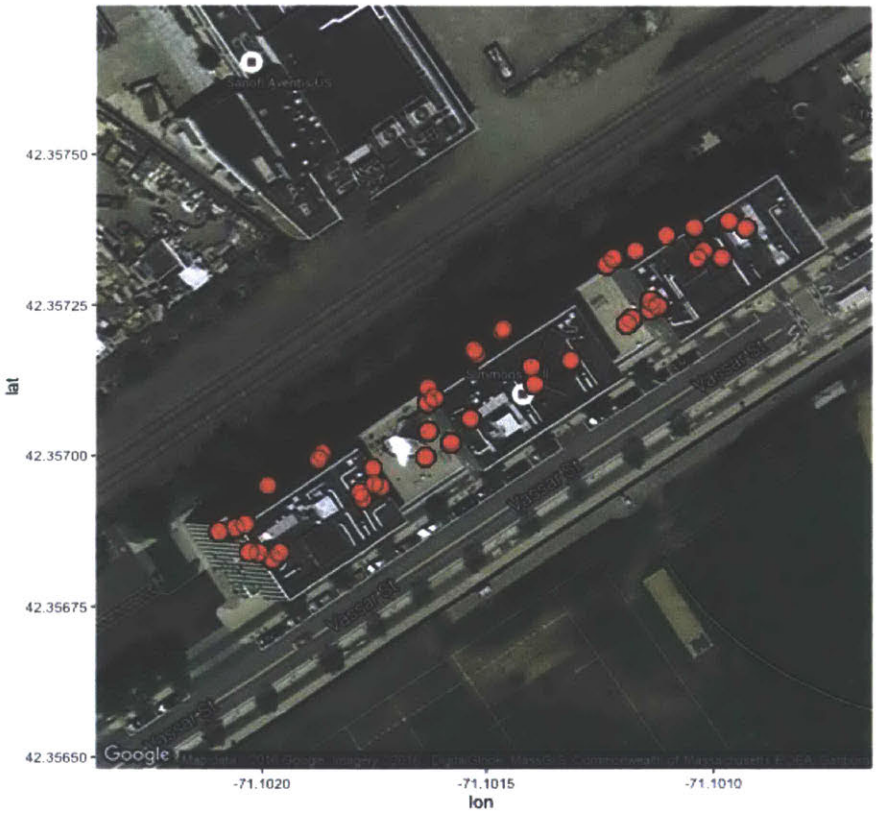


Figure 3.40 Distributed WiFi access points located within Simmons Hall.

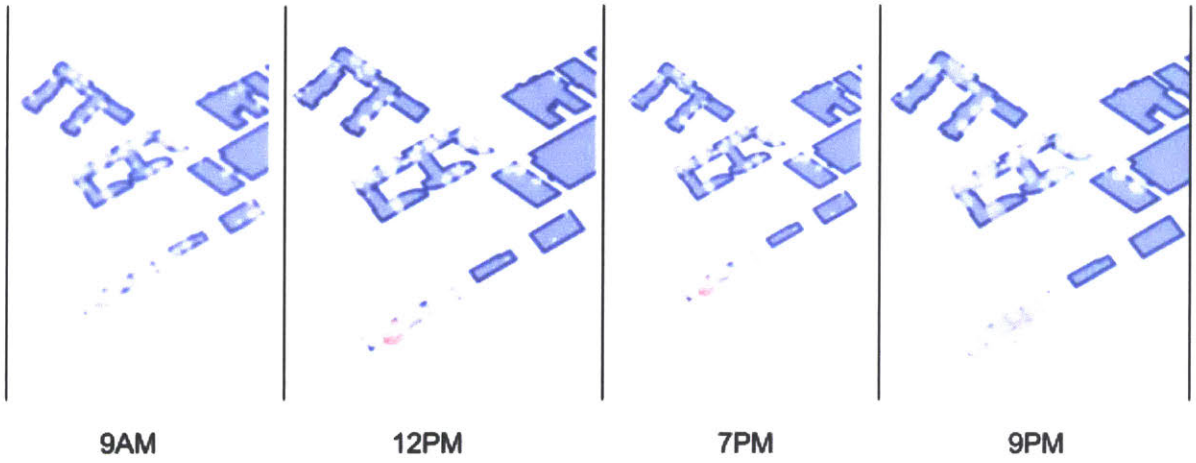


Figure 3.41 Simmons Wifi Activity throughout the day

Figure 3.42 Simmons Dining Hall

3.6 Analyzing Copley Square

So far, the buildings reviewed for analysis exist only at MIT and within the context of the MIT community. Another approach for comparison is to observe a place that is not at MIT and not a building. For this, the public open space, Copley Square in Boston, MA brings a different perspective from a design, management, and historical point of view: Where do people tend to congregate in a public square and how can this inform the design and placement of public furniture such as benches, tables, and other public furniture? What is considered an important feature of Copley Square through the public's eyes in multiple viewpoints? Can the geometric perspective yield discoveries and further analysis of art, architecture, urban layout and traffic patterns in the subjects that people photograph at such a location? Section 3.6.1 answers some of these questions.

Copley Square is one of Boston's busiest public spaces[15] whose landscape would evolve since its formation in the late 19th century of Victorian Boston. Notable buildings such as Old South Church (1873), Trinity Church (1877), the Boston Public Library (1895), the Copley Fairmount Hotel (1912), and the John Hancock Tower (1976) surround the square. Figure 3.43 shows a black and white photo of Copley Square in the 19th century. The square would go through a redesign phase in 1966 and later again in 1991 through an international design competition.

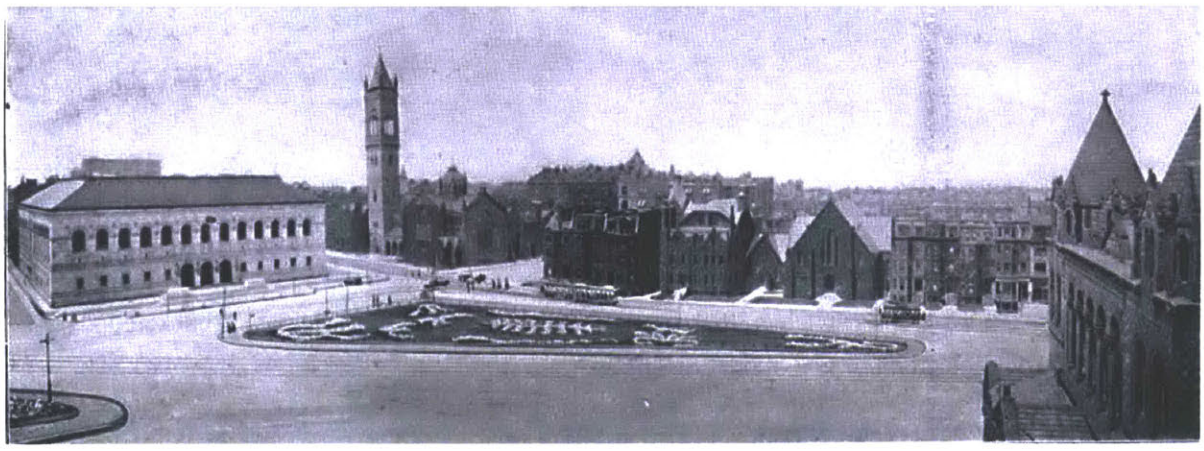


Figure 3.43 19th century Copley Square in Boston, MA.

[15] Tadisco, Patrice. *Copley Square*. (2012). *Landscape Notes*. Retrieved 22 September 2016, <https://landscapenotes.com/2012/08/23/copley-square/>

City planners and civic leaders had envisioned the new design to become a world-famous Boston landmark equivalent to St. Peter's Square in Rome. Landscape architects Sasaki, Dawson, and Demay Associates won the redesign competition in 1966. The new landscape suggests more of a plaza than a park with gradual concrete terraces that sunk twelve feet below grade (Figure 3.44) -- creating a visual disconnection from the surroundings. Budget cuts reduced the number of benches, trees, and other amenities in the original design which ultimately led Copley Square to become a desolate urban space and the backdrop for drug dealing and petty crimes.

Figure 3.44 Copley Square, Boston, MA 1970. General view with Boston Public Library in background (MIT Visual Collections, photograph by G.E. Kidder Smith).

On August 22, 1983, Mayor Kevin White announces the Copley Square Centennial Committee to serve as the public/private liaison for the renewal of Copley Square. Another competition takes place where the guidelines mention that the "square should function chiefly as a congenial

setting for conversation [and ease of public surveillance and control, as well as]... unplanned activities..only secondary should the Square be dependent for its animation on formally programmed events" (Todisco) In order to deter undesirable activities characteristic of desolate spaces, the new square requires seating for at least "1,000 people and 300 movable chairs, a demountable seasonal cafe, accommodation for a Farmer's Market and vendor's pushcarts were recommended" (Todisco).

To that end, Copley Square has become a vibrant community space or "outdoor living room" for the general public to enjoy till this day. With plentiful images of the modern Copley Square, do people form clusters in certain areas similar to Stata Center? When the keyword "Copley Square" is used for searching and collecting public images, what do individuals associate with the location? The Boston Public Library adjacent to the Square? Trinity Church situated on the Square? The Hancock Tower by architect I.M. Pei towering the landscape nearby or public events? Section 3.6.1 analyzes the original camera positions extracted from photographs of Copley Square to assist in answering the questions above.

The importance of knowing what people find intriguing about Copley Square in their photos give historians, city management, and designers the ability to know how people use urban spaces with its outdoor furniture and events thereby knowing what works and what does not work in the current urban scheme. Has crime reduced because the "new square" requires seating for at least a thousand individuals? Has tourism in the area increased because of public monuments located in the square? Do residents use and frequent the space? If so, for what purpose? Do the public events draw the community together in increased satisfaction, dissatisfaction, or neither? The next section takes a look at the types of photographs generated by the general public.

3.6.1 Copley Square as Photographed by the Public

As it turns out, the majority of individuals point their camera in the direction towards Trinity Church that was established on the site since 1733. Another dominant building within the frame and juxtaposed to Trinity Church is I.M. Pei's John Hancock Tower from 1977. The Boston Public Library, founded in 1848, sits opposite of Trinity Church, and the Fairmont Copley Plaza

Hotel from 1912 also has a dominant presence on the Square but these buildings do not draw attention to the public as evidenced in the image dataset. Figure 3.45 shows a sample of images captured by the public.

Figure 3.45 Publicly uploaded images of Copley Square.

Gathering the images to generate a 3D point cloud and extract the original camera position yields further evidence that the view facing Trinity Church is highly favorable amongst the crowd. Figure 3.46 shows a custom built 3D point cloud collected from the public images, where the majority of rendered points highlight Trinity Church as the main camera's focus. Notice that the architectural details of Trinity Church in the 3D point cloud contains much more detail compared to the surrounding buildings. More geometric features are brought by the number of images that focus on Trinity Church as the subject.

Figure 3.46 A 3D point cloud from scraping publicly uploaded images on the internet.

Extracting the original camera positions for each photograph produces further evidence that Trinity Church assumes the spotlight (Figure 3.47). Notice the density of cameras closer to the building and photographs taken from the center of the square and the slight change in angle of axis towards the left. (Figure 3.48). Overall, fewer cameras taper off towards each side of the square and towards the back. In other words, people prefer to take pictures of Copley Square close to Trinity Church and from the center, or slightly further away standing towards the left of the square pointing the camera from an angle. It is suspected that people photograph the square from left side so as to capture Trinity Church and the John Hancock Tower together within the frame, Figure 3.49 shows a sample of camera positions and its associated photograph.

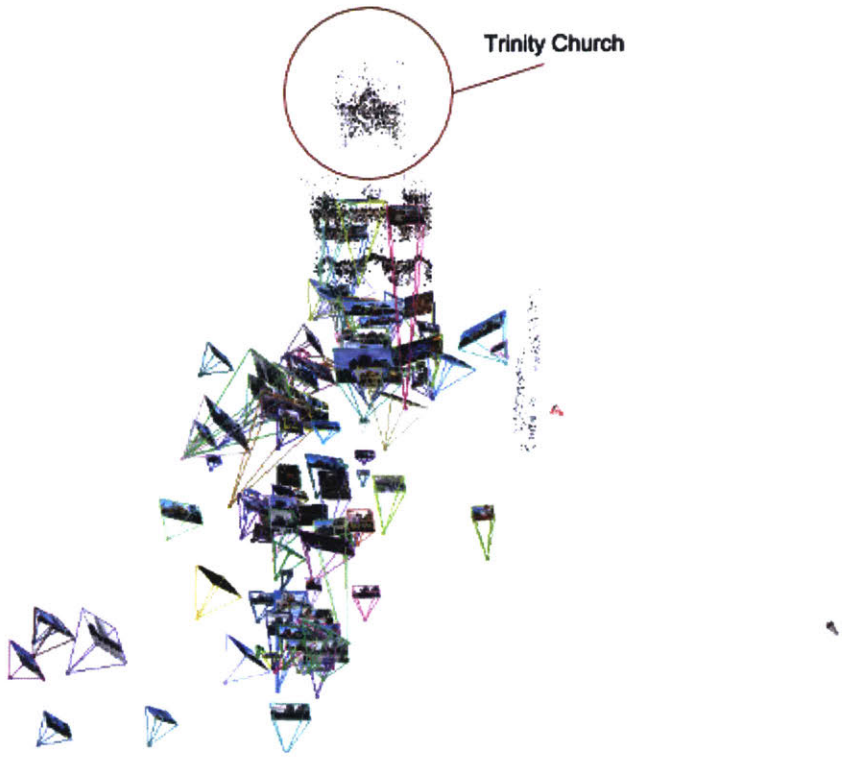


Figure 3.47 Original camera positions from public photographs pointing at Trinity Church. Superimpose map in photo.

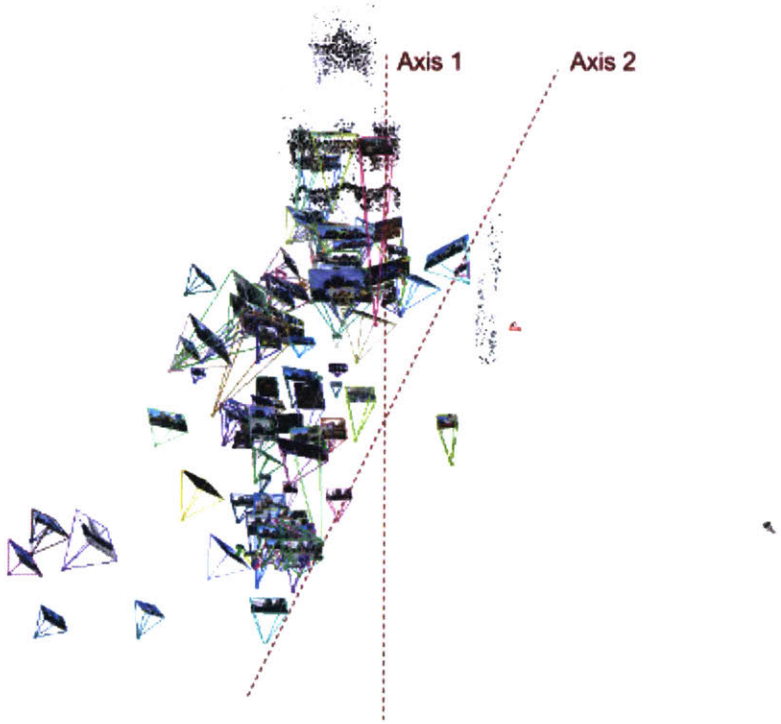


Figure 3.48 The sudden change from Axis 1 to Axis 2 when taking photographs of Copley Square.

Figure 3.49 Camera view angles and picture subjects.

3.7 Summary

Overall, the placement of camera positions vary based on access paths, however, despite there existing many access paths around a building/center of attraction, only certain facades or details of a spatial construct capture the public's attention to photograph as seen throughout the timeline within the image datasets. Another area for comparison would be to see how the public frame their images in comparison to professional photographs in publications -- to evaluate whether the general public and professional photographers capture similar icons of interest as deemed important.

Chapter 4

A Comparative Data Analysis of Stata, Simmons, and Copley

With plentiful public Application Programming Interfaces (APIs) hosted by organizations that allow users to programmatically query their servers, this chapter examines the possibility of mixing datasets outside of the ones already explored in the previous chapters. With MIT's WiFi data available, can we draw conclusions from exploring data from sites such as MeetUp.com? MeetUp is an online social networking site that facilitate groups to meet offline joined by shared interests. What about weather data? Do some weather patterns correlate to the presence and/or absence of individuals who photograph and visit MIT buildings? Do students remain in their housing quarters more than when inclement weather persists? Do images taken at Copley Square correlate to any offline social gatherings organized by a MeetUp group in Boston?

Mixing the datasets provide an expansion to the stories already told about the images, WiFi, and text data about MIT and Copley Square as explored on their own. If weather data and the presence of the number of connected devices on campus, for example, show that the MIT student community remain longer in their residences, then how can that information translate as to how MIT facilities manages energy usage on campus? How do MeetUps in Copley Square affect how future urban planning, or future redesigns of Copley Square several years from now? Can revealed crowd patterns extracted from today's web APIs inform how city planners impact stakeholders? Lastly, how does combining datasets from APIs retell the history and culture of a

place as seen through the lens of a historian? These questions shape the topics for exploration in this chapter.

4.1 Device Activity via WiFi Access Points

Location	Hours of low device activity	Hours of moderate device activity	Hours of high device activity
Stata Center	1a - 11a	10p - 12a	12p - 9p
Simmons Hall	2a - 9a	10a - 11a, 2p - 6p, 9p - 1a	12 - 2p, 7p - 8p

Table 4.1 Hourly Device Activity

The device activity around Stata Center and Simmons Hall differ partly due to the nature of the building's programming -- Stata Center's first floor is open to public and private events while Simmons Hall contains restricted access to the public, open only to dormitory residents. Table 4.1 shows the exact hours where low, moderate, and high device activity occur within the two buildings throughout the length of days in the WiFi dataset. The top diagram of Figure 4.1

illustrates Stata Center's hourly WiFi device activity for a 24 hour period within 4 days.

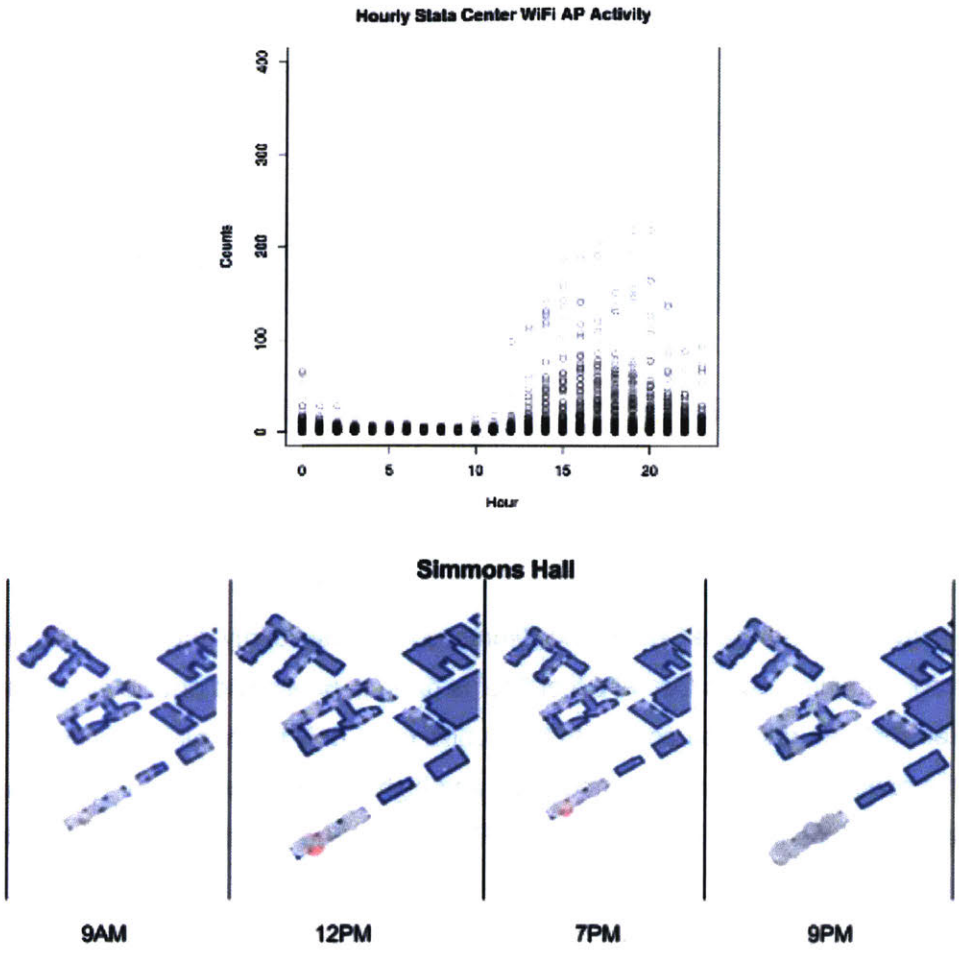


Figure 4.1 Cross comparison of WiFi device activity between Stata Center and Simmons Hall

Between Stata and Simmons, the flow of crowd activity differ in that Stata tends to become crowded with foot traffic beginning at noon and wanes starting at 7pm up until midnight. Simmons, on the other hand, shows a pattern of high traffic activity in the hour of noon and 7pm while the rest of the day shows neutral activity. The high traffic pattern occurs toward the west side of the building -- the dorm's cafeteria location.

4.2 Text Data

A side by side comparison of text data between Stata Center, Simmons Hall, and Copley Square show that individuals largely describe the building by their given name and then further use words to express the building in very different ways. Stata Center, for example, is described as a "childlike," "whimsical," "surreal" building, whereas "superfav," "fav," and "architecture" is used for describing Simmons Hall amongst other words. Copley Square currently contain words that describe more recent events such as the Boston Marathon Memorial. Table 4.2 shows the frequency of words used and Figure 4.2 shows a thorough frequency chart of terms.

Location	Most frequent words	Least frequent words	Unique descriptive words
Stata Center	Frank Gehry, Cambridge, Stata Center	Explore, build	Whimsical, childlike, surreal, gehryeffect, gaudi
Simmons Hall	Architecture, architect, architecture contemporary, american insttute of architecture	Cloud, sponge	Superfav, fav, architecture
Copley Square	Copleysquare, massachusetts, backbay, boston	Addition, africanamerican, airplane, alley	Bostonstrong, church, engagements, bostonbombingmemorial, citylight, concert

Table 4.2 Cross comparison of word usage to describe Stata, Simmons, and Copley.

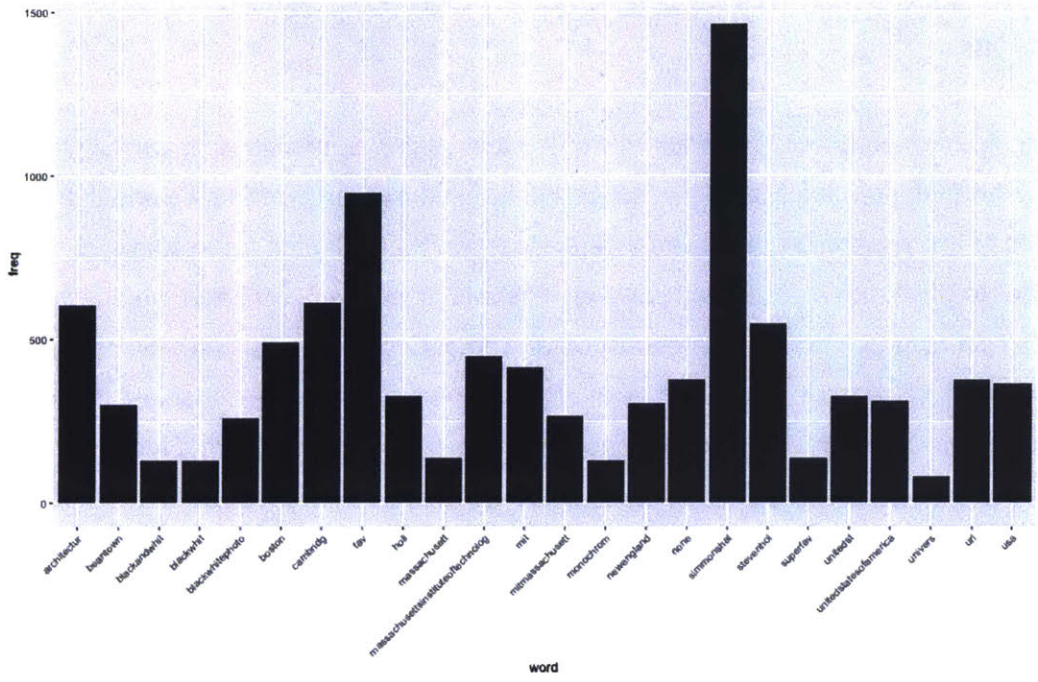


Figure 4.2 (a) Simmons Hall photo tags.

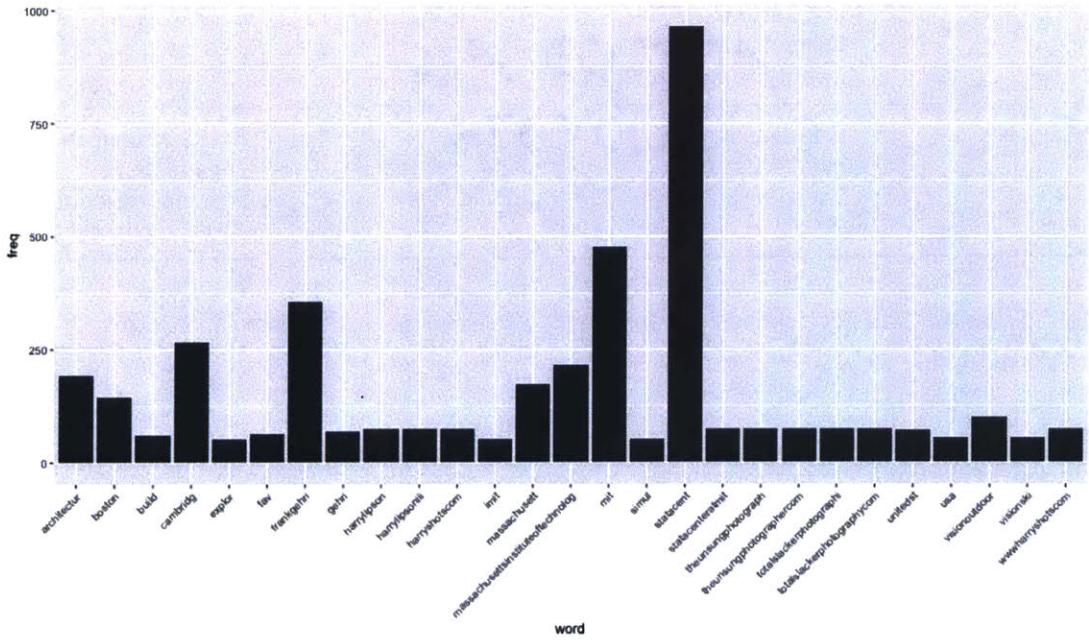


Figure 4.2 (b) Stata Center photo tags

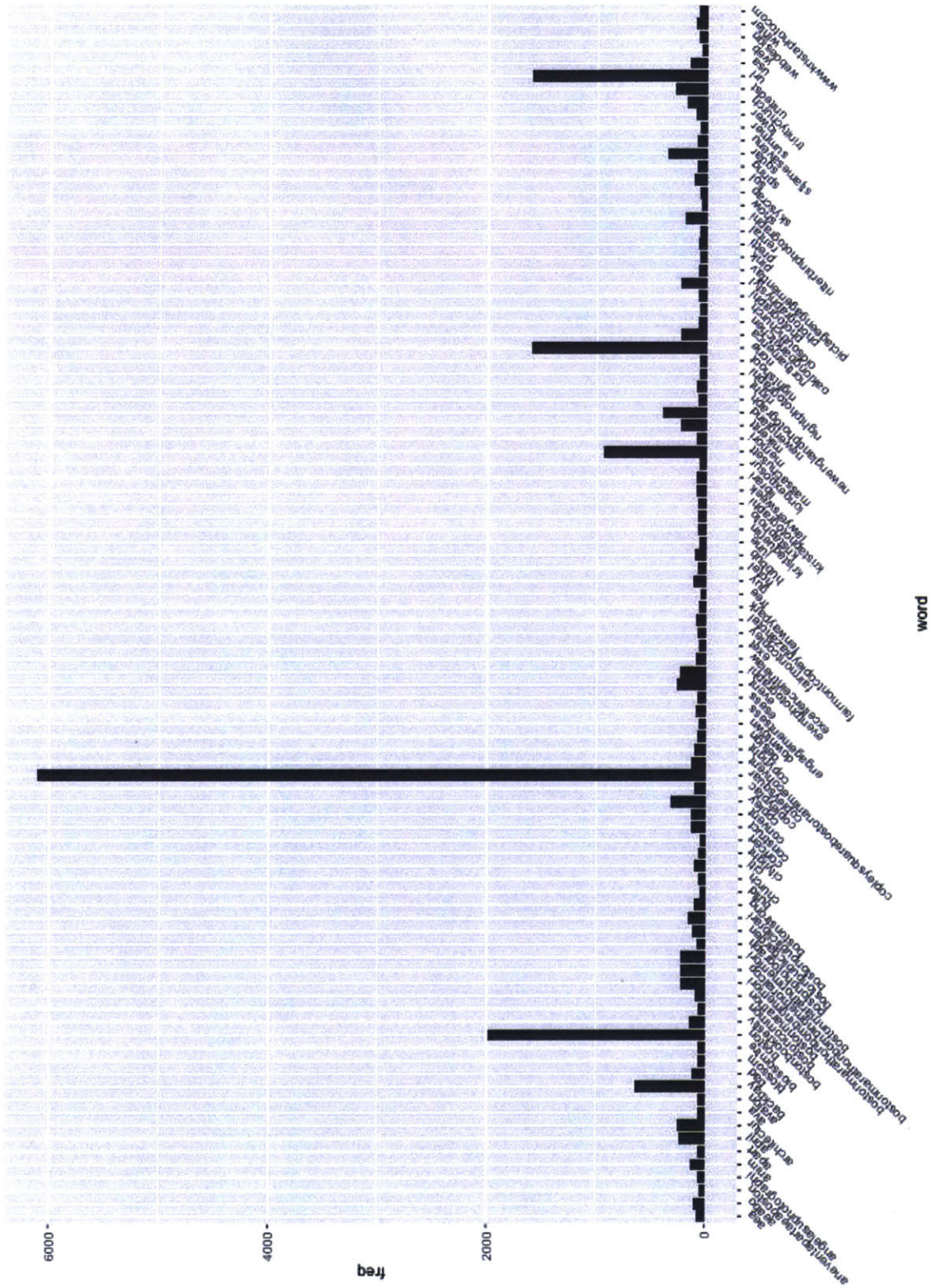


Figure 4.2 (c) Copley Square photo tags.

4.3 Time Series Summary

Location	Winter	Fall	Spring	Summer
Stata Center				
Simmons Hall				
Copley Square				

Low density

High density

Moderate density

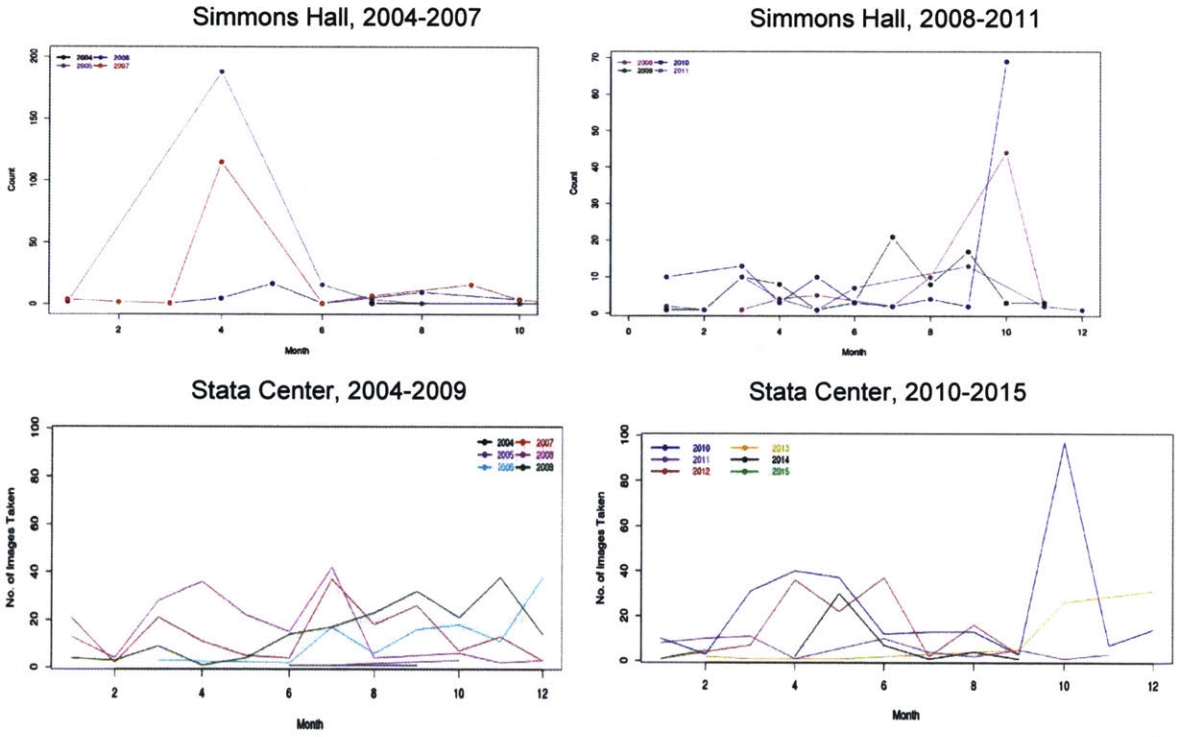


Table 4.3 Population density of Simmons Hall, Stata Center, and Copley Square throughout the year.

Between Stata Center, Simmons Hall, and Copley Square, these locations show a similar trend in building visitation - an ebb in activity occurs in the winter months and then a rise and slight dip pattern in the Fall, Spring, and Summer months (Table 4.3). No matter the attraction type,

the three locations are subject to New England's seasonal weather patterns and consequently affect the number of attendance at each location. The data for MIT confirms the observations in increased/decreased visitor attendance as mentioned by staff at the MIT Information Office. The next section summarizes the types of cameras that visitors bring with them when visiting an attraction, such data reveals some demographic information about the visiting population at MIT and Copley Square.

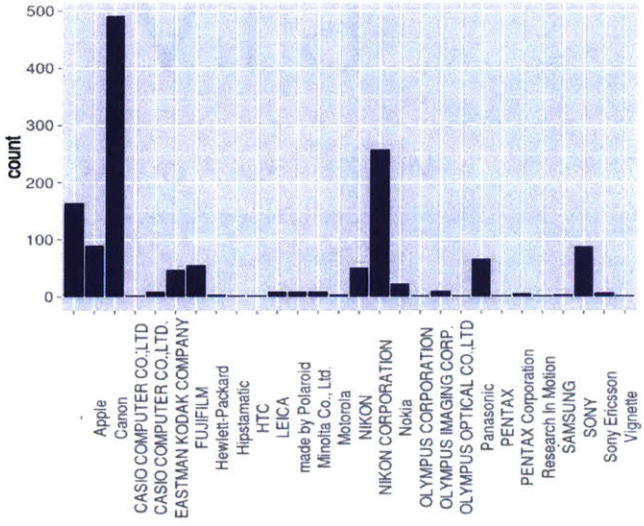
4.4 Camera Types

For each location, the type of cameras used for photographing buildings remain similar -- professional Canon and Nikon cameras, and occasionally a mobile device such as an iPhone (Table 4.4). The Canon and Nikon camera models reveal that the type of user who engage with such cameras value composition, lighting, and other technical photographic details to create an image. In this case, assumptions can be made about the type of visitors who populate Stata Center, Simmons Hall, and Copley Square in the given dataset -- professional photographers and/or hobbyists. The dataset, however, comes from Flickr which is a public image database that is mostly frequented by professional and amateur photographers to share and display their works.

Table 4.5 illustrates the top three camera types that visitors bring to Copley Square, Stata Center, and Simmons Hall. A variety of Canon and Nikon camera models are used to photograph Simmons Hall and Stata Center (see Appendix D for a thorough list of camera types and frequency usage on each building site) despite the large existence of camera equipped mobile phone devices. Based on an empirical observation and the proliferation of social networking sites such as Facebook and Instagram, active social networking mobile device users upload photographs onto the mentioned platforms. As mentioned in later chapters, such platforms, strip the metadata from images for privacy reasons. The metadata serves as the necessary component in order to extract the camera types and other image data. In this case, the Flickr database contains preserved image metadata and becomes the lense for observing the camera type patterns found in this part of the research.

4.5 Camera Positions

Extracting the original camera positions from each photograph for Simmons, Stata, and Copley Square reveal where individuals crowd around buildings and plazas. Table 4.6 shows a side by side comparison for each location, revealing that Stata Center has many pedestrian paths but visitors choose to stand in areas A, B, and C over time, since 2004. Simmons Hall, on the other hand, stretches along the east and west side where individuals occupy most of the road and grassy field areas -- the site and building configuration determines where people form into crowds. The majority of camera positions occur on the west side of Simmons Hall highlighted as Region B in Table 4.6, then the east side (Region A) and grassy fields (Region C) stand as the second and third popular locations to situate a camera. As for Copley Square, individuals stand along Axis 1 and shift midway within the square to Axis 2 and leave most of the courtyard empty. Otherwise, the majority of camera positions remain concentrated in one central region labeled as Region A in Table 4.6.

Location	Camera Types
Stata Center	

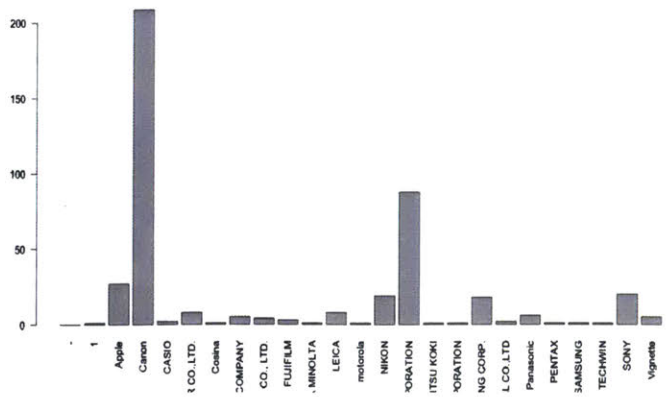
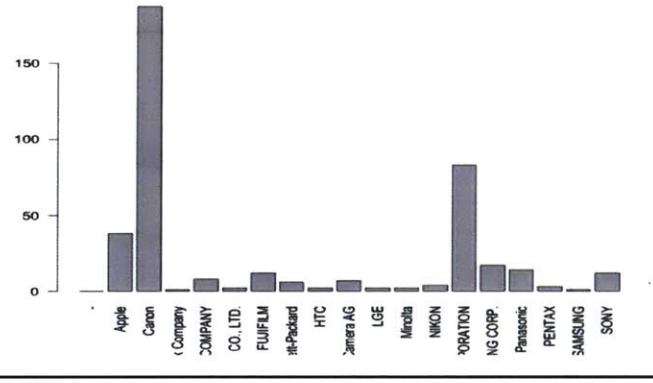
Simmons Hall	
Copley Square	

Table 4.4 Devices used to photograph across Simmons Hall, Stata Center, and Copley Square.

Building	Nikon	Canon	Apple
Stata Center	307	491	89
Simmons Hall	90	208	27
Copley Square	107	208	27

Table 4.5 The top three camera types that visitors bring to each site.

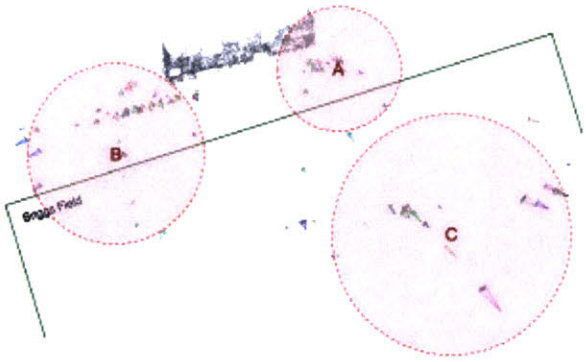
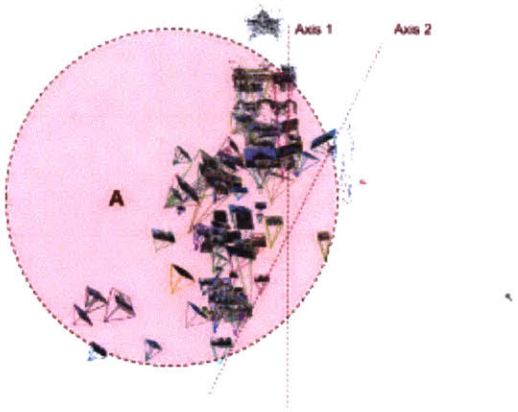
Location	Camera Positions
Stata Center	
Simmons Hall	
Copley Square	

Table 4.6 Varying camera positions for each site.

4.6 Mixing Data: Using the Weather API

How does weather factor with the arrival of tourists to Boston and the number of photographs they take of landmarks? Figure 4.3 illustrates the extraction of dates from the Stata Center image dataset shown as a scatter plot along with the average temperatures in Boston on those dates. The curve drawn on the graph is a locally weighted scatterplot smoothing method (Loess Curve) that shows the typical pattern for Boston throughout the years. As the scatter plot indicates, more photographs are taken when temperatures are closer or above the curve, then the number of data points wane at and below 30 degrees Fahrenheit. In terms of WiFi connectivity and outside temperatures, Figure 4.4 shows that the highest temperatures were reached on September 28 and lowest temperature on September 30. Integrating other API datasets can provide further information about the buildings and locations examined, Appendix E illustrates the extraction of streaming MeetUp API data for Cambridge and Boston.

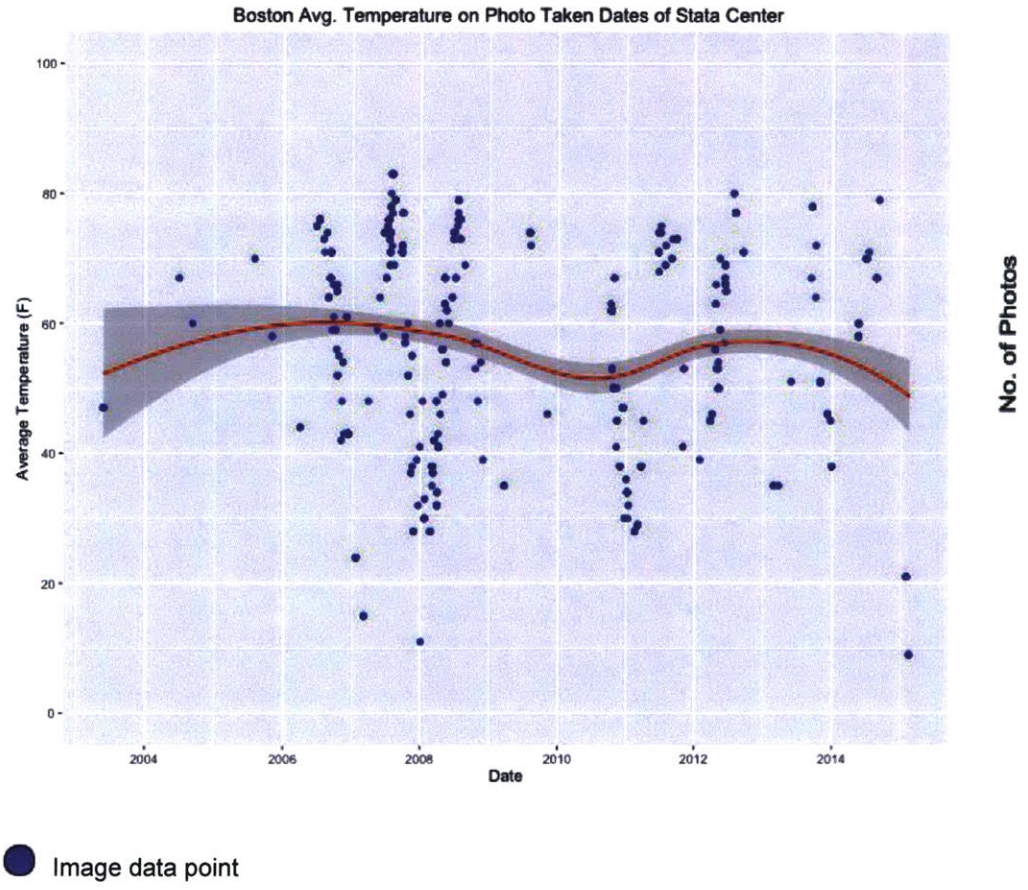


Figure 4.3 Temperature on days Stata Center was photographed by the general public.

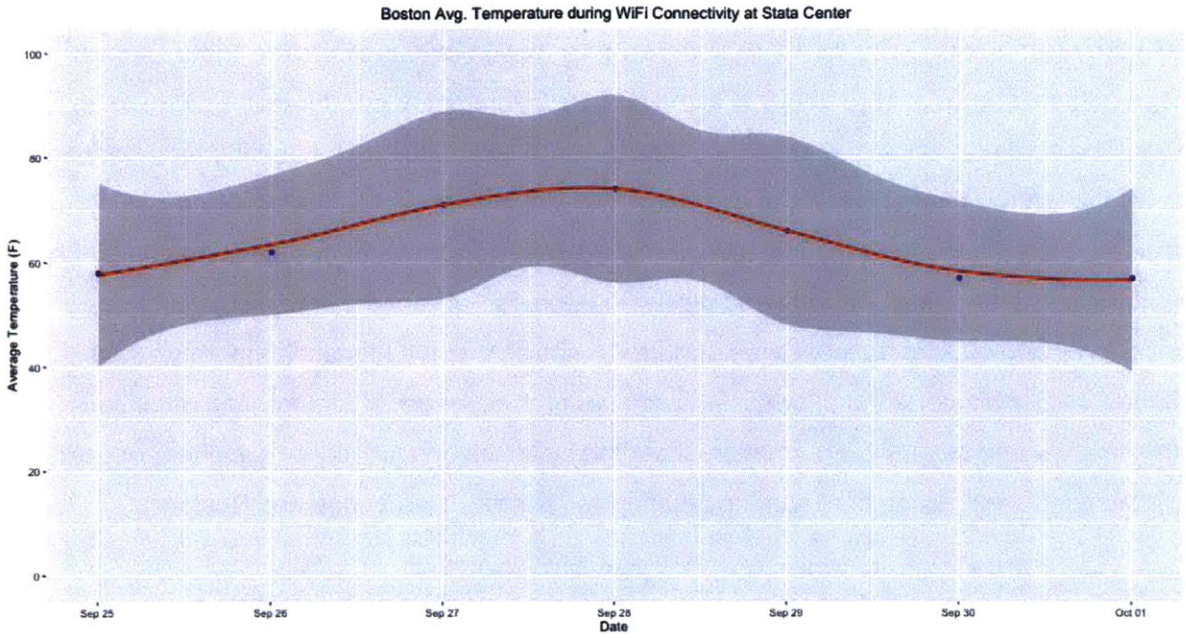


Figure 4.4 Weather temperature during the selected dates of WiFi connectivity at Stata Center

Part II. Research Methods and Algorithms

Chapter 5

Exploring Algorithms and Models in Computer Vision, Machine Learning, and Statistical Reasoning

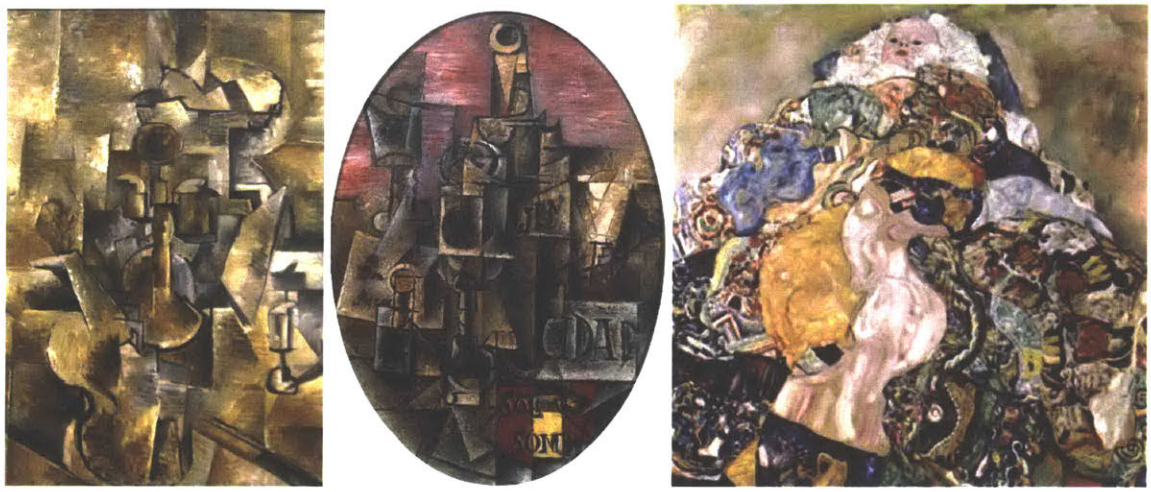
5.1 Introduction

In 2014, it was reported that a machine learning algorithm could classify fine art paintings from the early 15th century to the late 20th century and found a previously unrecognized influence between artists[16] [17]. To find the influence between artists, art experts compare space, texture, form, color, shape, and other attributes. Similarly, the algorithm also constructs a comparison of attributes by using 1700 paintings by 66 artists in 13 different styles as its training dataset, or a dataset used as ground truth. The algorithm found influences that were already known by experts such as how the early 20th century Austrian painter Gustav Klimt's was influenced by artists George Braque and Pablo Picasso (Figure 5.1). The paintings that had not been identified by art experts but was found by the algorithm, was the influence on Norman Rockwell's *Shuffleton's Barber Shop* (c. 1950) to Frederic Bazille's *Studio 9 Rue de la Condamine* (c. 1870) in Figure 5.2. Upon visual inspection between the paintings, Rockwell's

[16] "When A Machine Learning Algorithm Studied Fine Art Paintings, It Saw Things Art Historians Had Never Noticed". The Physics arXiv Blog. Accessed 23 August 2015.
[17] Babak Saleh, Kanako Abe, et al. "Toward Automated Discovery of Artistic Influence." Computer Vision and Pattern Recognition. Cornell University. Vol 1408.3218, 2014.

painting shares similar compositional styles to Bazille's -- the placement of human subjects nearly two-thirds of the canvas, a large window that draws a viewer's eye to the activities of the human subjects, and the compositional placement of similar objects between the scenes such as the chair and stove heater.

Classifying paintings and finding stylistic influences between artists demonstrate one of the applications of machine learning. In other applications, machine learning algorithms have become a ubiquitous part of our real-world applications -- airlines use prediction algorithms to



(a) (b) (c)

Figure 5.1 (a) George Braque's Man with a Violin, (c. 1912). (b) Pablo Picasso's Violin and Candlestick, (c. 1912). (c) Baby (c.1917)

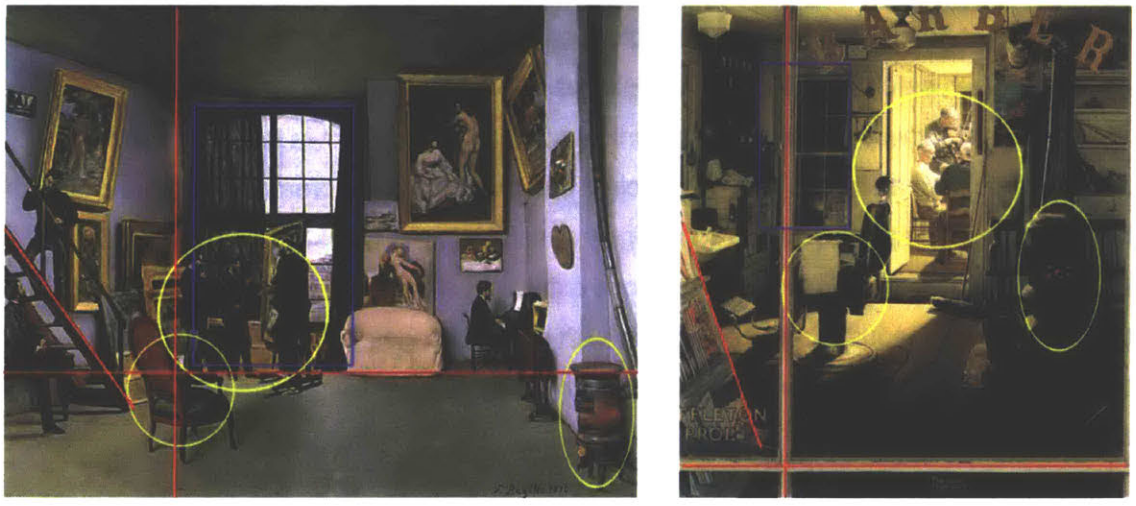


Figure 5.2 (left) Frederic Bazille's *Studio 9 Rue de la Condamine* (c. 1870). (right) Norman Rockwell's *Shuffleton's Barber Shop* (c. 1950)

determine flight delays that save travelers and the airline time and monetary resources, online retail websites feature product recommendations relevant to each visitor, banks build up a picture of how people spend, spam filtering automatically sorts irrelevant emails from relevant ones in our inboxes, and the list continues.

This chapter looks into the choice of algorithms and statistical models for implementation in this research, but before going further, it's useful to step back and ask: What does it mean to learn? Learning a new task requires observation and building upon previous knowledge, if any. When humans make mistakes, they try something different to correct their course until achieving the desired output. Just like the human learning process, machine learning is about learning from data, recalling previous knowledge, and adapting its course of action to a task. Other areas of intelligence include reasoning and logical deduction, but this chapter and research is only interested in the fundamental aspects of learning as described above.

5.2 Computer Vision

One of the challenges in this research is dealing with many photographs data mined from Flickr that do not match the criteria for an optimal image dataset of Stata Center, Copley Square, and Simmons Hall. The use of the term optimal is meant to describe an image dataset that purely contains photos of only exterior building facades and not images of people on location or incorrect labeling of buildings (i.e. there's a picture of Stata Center in the dataset mistagged as Simmons Hall). The usefulness of examining images of people and events can be useful as a study in and of itself, however, one of the technical goals for this project is to determine the original camera pose of an image and then project the image pixels to create a 3D point cloud model of a building. The process to do so is highly sensitive to the types of images fed as an input into the 3D reconstruction and camera pose extraction algorithm. So, mixing images of people, events, or erroneously tagged buildings with correct images of building facades can construct a very messy output. Chapter 6 will go into detail about the challenges to overcome that very process.

The process to create a 3D image-based model and the extraction of images require a five part process: 1) Match similar images; 2) Learn from matched images; 3) Classify correct from

incorrect images of a building; 4) Find the camera pose; 5) Project image pixels for 3d reconstruction. This section will explain each algorithm in that five part process.

5.2.1 Scale Invariant Feature Transform (SIFT)

One of the first steps to semi-automatically separate incorrect tagged images of buildings from the correct image dataset, is to help the computer understand the attributes of each picture -- pixel density, structural outline, and others discussed in this section. To let the computer understand images, the Scale Invariant Feature Transform (SIFT) algorithm accomplishes just that. The algorithm was developed in 2004 by D.Lowe of University of British Columbia where prior to SIFT, algorithms for image matching were neither rotation nor scale invariant. Said another way, if we want to locate an object in an image containing many other objects, SIFT is not affected by scaling or rotation of objects in an image like in previous methods.

SIFT extracts feature keypoints in an image, each keypoint detection contains attributes such as its (x,y) coordinates, angle of orientation, and size of neighborhood. Each pixel is then compared with its 8 neighbors at different scales and its edges are computed, from there each edge is searched for dominant attributes -- this process is called the Difference of Gaussians[18]. Matching each image's keypoint with another image allows the computer to identify identical sets (Algorithm 5.1). Figure 5.3 shows an example of extracting keypoint features from MIT's Ray and Maria Stata Center.

Algorithm 5.1 SIFT

```
Input: load image

for each octave
      create Guassian blur
      create difference of Guassian intervals
      compute edge

search each octave for "stable extrema"
```

[18] SIFT - OpenCV documentation
http://docs.opencv.org/trunk/doc/py_tutorials/py_feature2d/py_sift_intro/py_sift_intro.html

```
create keypoints at dominant orientations

for each keypoint
     rotate sample grid to keypoint orientation
     sample region and create descriptor

save pyramid images
save output images
save descriptors
```

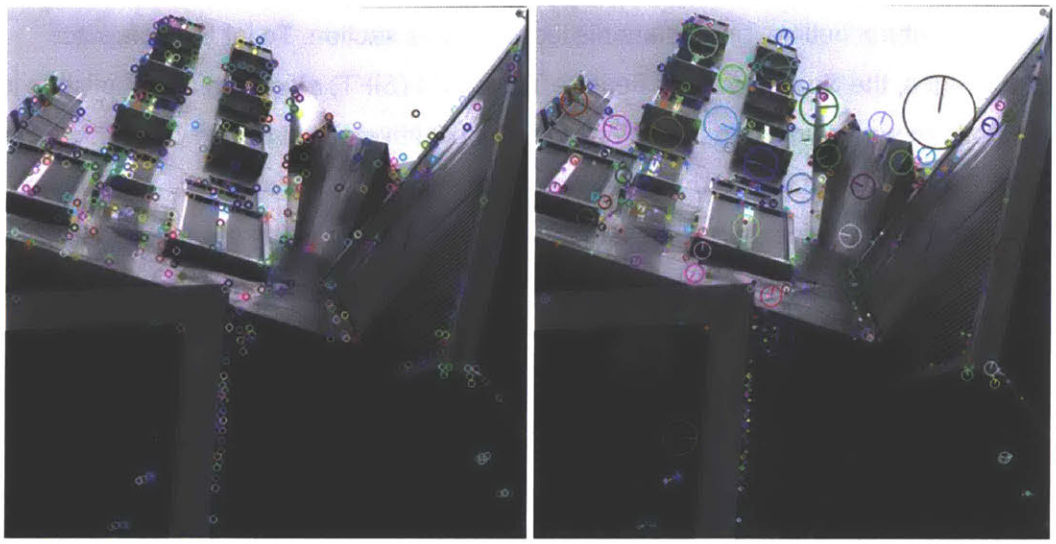


Figure 5.3 (right) Results of the SIFT algorithm identifying keypoint descriptors. (left) Identifying keypoint descriptors and vector orientation.

After identifying the keypoint descriptors for each image, the SIFT algorithm performs a matching process. Figure 5.4 shows the match between two images of Stata Center. The process of SIFTing can produce robust results though not without some errors as seen in Figure 5.4. Image "one" contains 1,180 features and image two has 1,940 features, matching 25 features and 8 inliers. Notice that the glass doors in image "one" does not correspond correctly to image 2 and instead matches the glass with the lawn in image "two." Actually, the algorithm does not perfectly match the images. To solve for these errors, we can use the Affine SIFT approach.

Image 1 Image 2

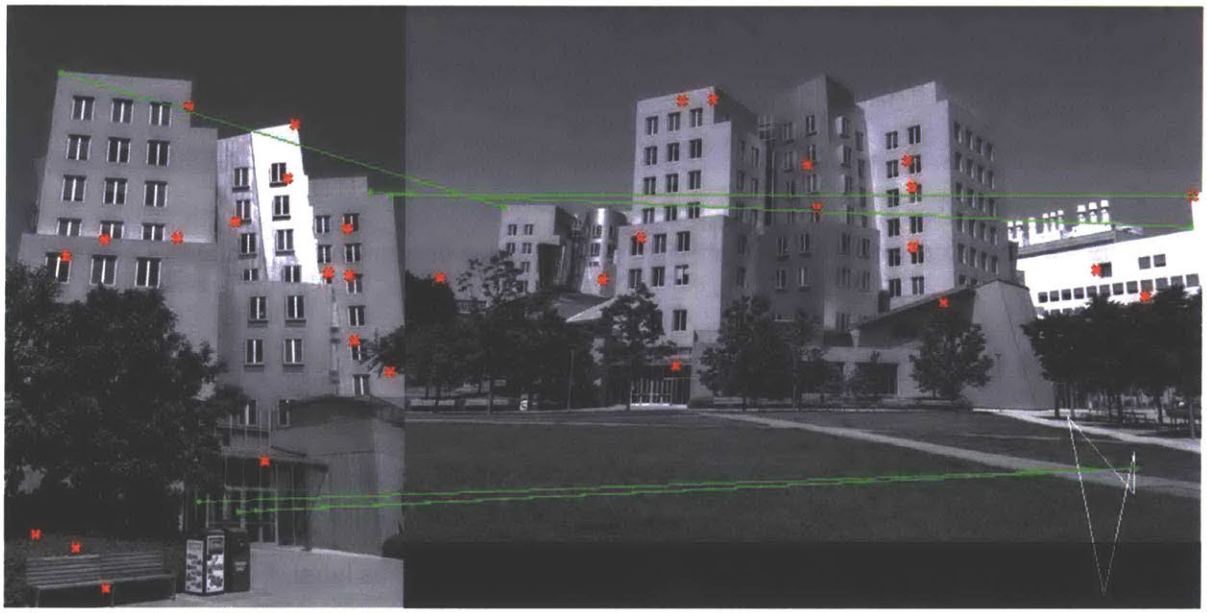


Figure 5.4 SIFT object detection

Figure 5.5 produces less errors using an affine invariant SIFT image matching algorithm. Unlike the previous algorithm, the affine invariant algorithm found 23,384 features in image 1 and 35,766 features in image 2 with 29 inliers and 310 features matched. the feature and matching points. The affine detector applies a "set of affine transformations to the image, detects keypoints, and re-projects them into initial image coordinates." ASIFT: An Algorithm for Fully

Affine Invariant Comparison was first published in 2011 by Gushen Yu and Jean-Michel Morel. When the view angle changes, SIFT fails and ASIFT works.[19]

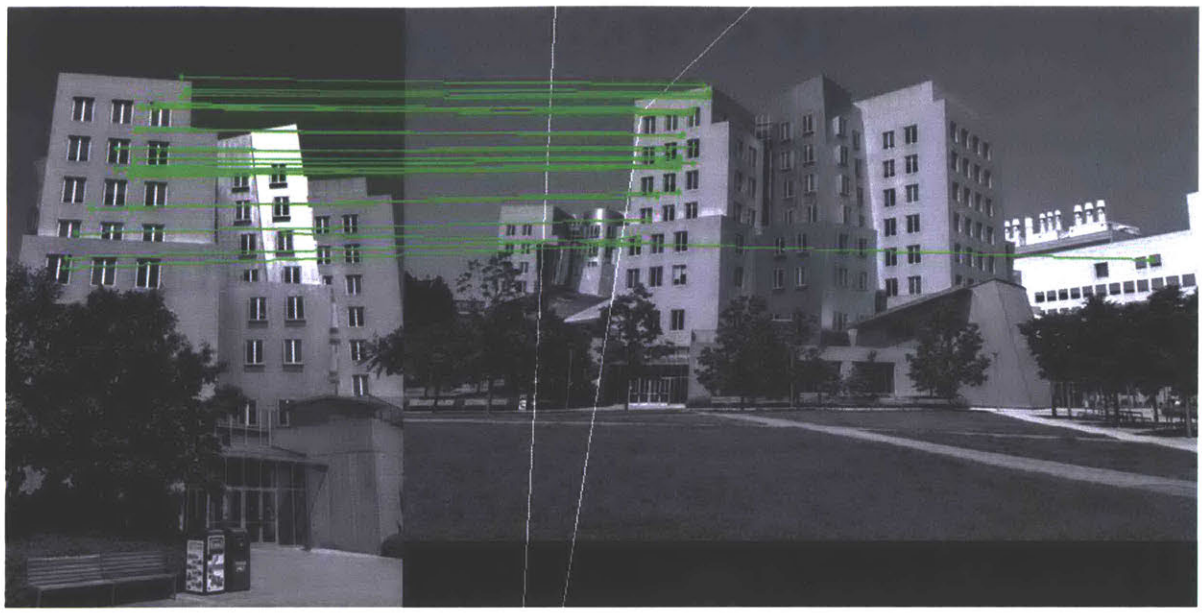


Image 1 Image 2

Figure 5.5 Less errors with affine invariant feature-based image matching between images 1 and 2.

5.2.2 Support Vector Machine for Classification

Support Vector Machine (SVM) is a typical classification algorithm that tries to find a hyperplane that separate features and labels in a dataset. It is about learning structure from data by mapping $X \rightarrow Y$ where $x \in X$ is some object and $y \in Y$ is a class label. For example, in a two-class classification problem, $x \in R^n$, $y \in \{\pm 1\}$, we want to distinguish whether a photograph is an indoor or outdoor image of Stata Center in a set of five hundred photographs. So, given a previously seen $x \in X$, we need to find a suitable $y \in Y$. To accomplish this, we provide a training set and create prediction models where the input/output sets X, Y and training set $(x_1, y_1), \ldots, (x_m, y_m)$ produce a generalization or learns a classifier: $y = f(x, \alpha)$ where α are the parameters of the function. The right diagram in Figure 5.6 shows how an SVM linearly separates the data features by a hyperplane thus creating a decision boundary. SVM provides

[19]Guoshen Yu, and Jean-Michel Morel, *ASIFT: An Algorithm for Fully Affine Invariant Comparison*, Image Processing On Line, 1 (2011).http://dx.doi.org/10.5201/ipol.2011.my-asift http://www.ipol.im/pub/art/2011/my-asift/

advantages that can outperform linear classifiers by its capability to also create non-linear decision boundaries for n-class categories using kernels (Figure 5.6 left)

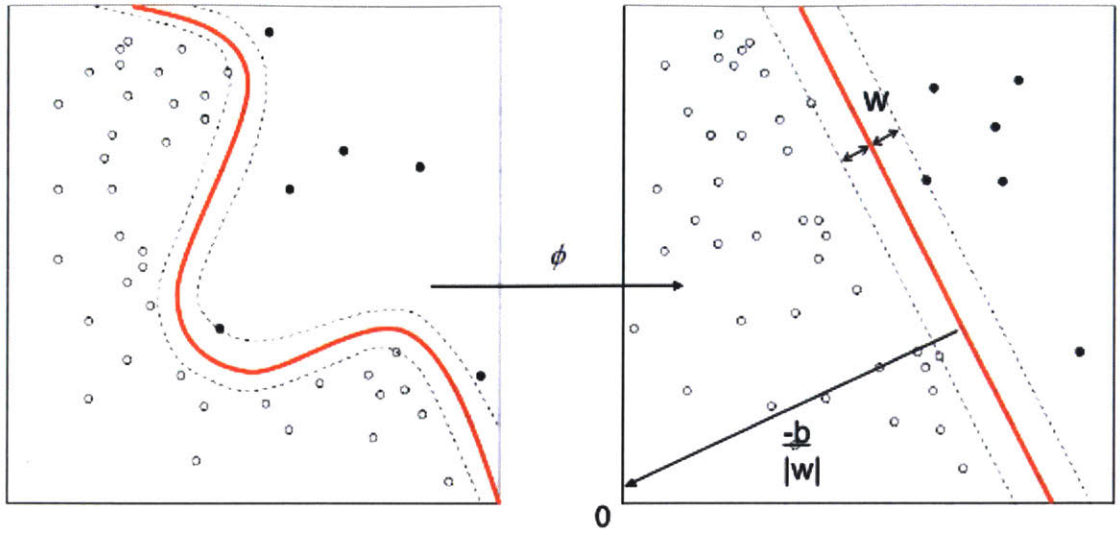


Figure 5.6 (Left) Support Vector Machine non-linear classifier (Right) and SVM linear classifier of two separable classes

For a linearly separable case, the margins to the left and right of the red line/polyline of Figure 5.6 is a hyperplane and is normal or perpendicular to the red line. The hyperplane can be described as $\mathbf{w} \cdot \mathbf{x} + b = 0$ where:

- **w** is normal to the hyperplane
- $\frac{b}{|w|}$ is the distance that is perpendicular from the hyperplane to the origin

SVM essentially is about selecting variables **w** and b so that the training data can be described by:

$$x_i \cdot w + b \geq +1 \quad \text{for } y_i = +1 \qquad \text{(Eq. 3.1)}$$

$$x_i \cdot w + b \leq +1 \quad \text{for } y_i = -1 \qquad \text{(Eq. 3.2)}$$

Combining the equations become:

$$y_i(x_i \cdot w + b) - 1 \geq 0 \; \forall_i \qquad \text{(Eq. 3.3)}$$

The strategy for semi-automatically categorizing images with SVM, SIFT, and another technique called Bag of Visual Words (described in the next section) follows a solution proposed by Gabrielle Csurka et. all in their paper "Visual Categorization with Bags of Keypoints" as shown in Algorithm 5.2

Algorithm 5.2

```
Compute SIFT descriptors at keypoints
Group similar descriptors into clusters
Compute training data for SVM classifier

for each image
     Compute SIFT descriptors
     Associate each descriptor into the cluster by Euclidean distance
     Create histogram from the associated image to descriptor

for each image category
     Train classifier with positive examples
input unseen images

for each unseen image
    Classify image using trained SVM
    Use SVM to sort image and return confidence predictor
```

5.2.3 Bag of Visual Words

The images scraped from social media sites often contain desired images alongside undesired images. For example, downloading 1000+ images of Frank Gehry's Stata Center will not only include pictures of the building's exterior, but also images of the building's interior, images with people, or objects inside of the building. This becomes problematic when we need to determine the camera's pose when performing a 3D reconstruction with multiple view geometry algorithms on an unordered set of images. See Figure 5.7.

Figure 5.7 Example of Stata Center images returned from scraping Flickr.

To solve this, the use of machine learning can correctly classify, with a margin of error, what set of images will belong to the specific category, Stata Center versus Simmons Hall, for example. The bag of words model is a popular technique for image classification and is also used in natural language processing. To classify images, the algorithm must undergo a training phase to determine which class an unseen image will belong to. Initial test images were explored during the undertaking of this research before finalizing the image dataset to Stata Center, Simmons Hall, and Copley Square. The test images of buildings in the initial training set consist of Stata Center by Frank Gehry, Baker House by Alvar Aalto, and the Harvard Carpenter Center for the Visual Arts by Le Corbusier. Each of these categories belong to separate classes and comprise of the training dataset, Figure 5.8a.

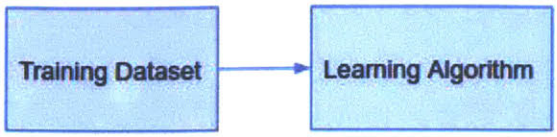


Figure 5.8 (a) Images are trained and fed to the learning algorithm

The learning phase involves the following steps:

1. Select images and extract local features, or keypoints and descriptors for each image using the SIFT algorithm.

2. Compute the visual words via k-means (described in the next section). Each keypoint belongs to a cluster to form the bag of visual words. Find similar image descriptors between clusters where the euclidean distance between the SIFT descriptors are clustered into k number of groups using the K-Means algorithm.

3. Group the histograms of visual words for each training image.

4. Feed the histogram to the classifier for training a model.

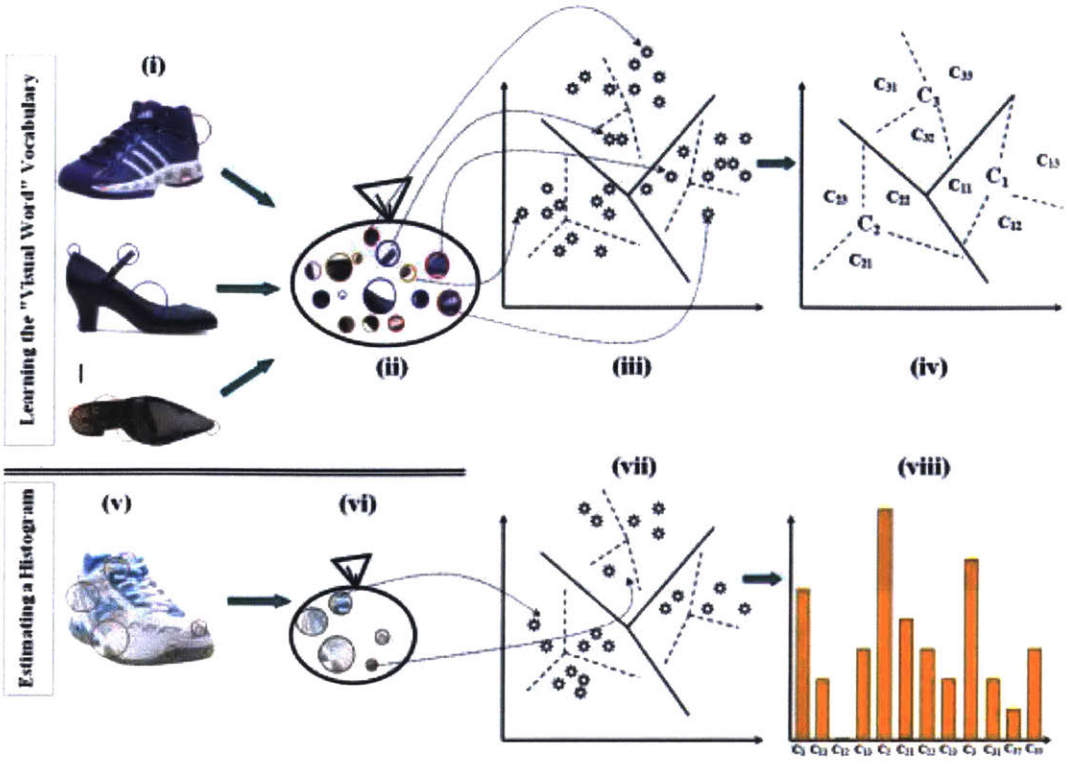


Figure 5.8 (b) Bag of Visual Words[20]

Figure 5.8b shows how the Bag of Visual Words model creates a visual vocabulary. In part (i), SIFT features are extracted from each shoe category, (ii) Each feature is placed into the bag of visual words, (iii) the k-means algorithm finds a separation by category, (iv) Set the visual vocabulary, (v) Extract SIFT features from new image, (iv) Obtain SIFT feature points to put into bag of visual words. Part (vii) Matches feature descriptors with previous visual vocabulary to ultimately break down similar features into a histogram in part (viii). After the Bag of Visual Words model is complete, the next stage involves passing the histogram to the SVM algorithm. In this case, after instantiating the learning algorithm for three example classes (Stata, Baker House, Carpenter Center) and computing the k-means cluster, each image is then assigned a codified label (see Figure 5.9 and Table 5.1) to represent a category.

[20] Image from "Tagging Products using Image Classification" Brian Tomasik, Phyo Thiha, and Douglas Turnbull, SIGIR 2009

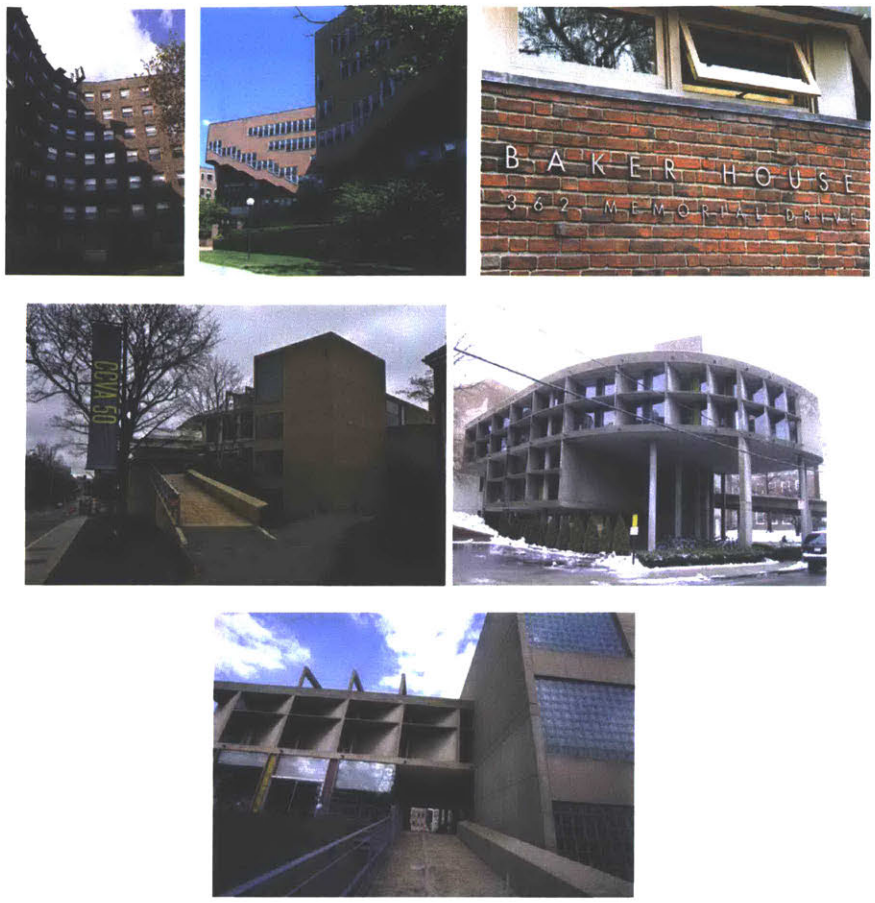


Figure 5.9 Sample images of MIT's Baker House and Harvard's Carpenter Center

Category	Label
baker	0
carpenter	1
stata	2

Table 5.1 Codified labels for each image category

The classification process involves loading an unseen image to compute the visual words histogram, write the histogram to file to pass it to the SVM, and tests the data with the SVM algorithm. The output prediction is from *n...n+1*. The learning algorithm passes the histogram

taken from the bag of visual features in the training set so that the classification algorithm can use the histogram and cluster analysis upon the test data set (of unseen images). The histogram counts all the similarities and places the features in bins. The following section explains k-means clustering as a means for grouping similar features for each image.

5.2.4 K-means Clustering

The k-means clustering groups each observation (image descriptors, in this case), $x_1, x_2, ...x_n$ by partitioning n observations into k $\leq$ n sets S = $\{S_1, S_2, ...S_k\}$ and then taking the sum of the difference of each observation along with its mean within a set:

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2 \qquad \textbf{(Eq. 5.4)}$$

Equation 5.4. Squared error function where S_i = 1,2...k and μ_i is the mean point of all points x_j in S_i

As illustrated in Figure 5.10, each observation is sorted into three separate classes. The circle with an "x" is the cluster's mean point. The advantage of k-means clustering provides computationally faster results as well as tighter clusters. The disadvantage is that the technique does not work very well if there are non-globular clusters, or clusters that do not have well defined center points. The cluster can also result in a different partition, re-running the algorithm helps to compare the results.

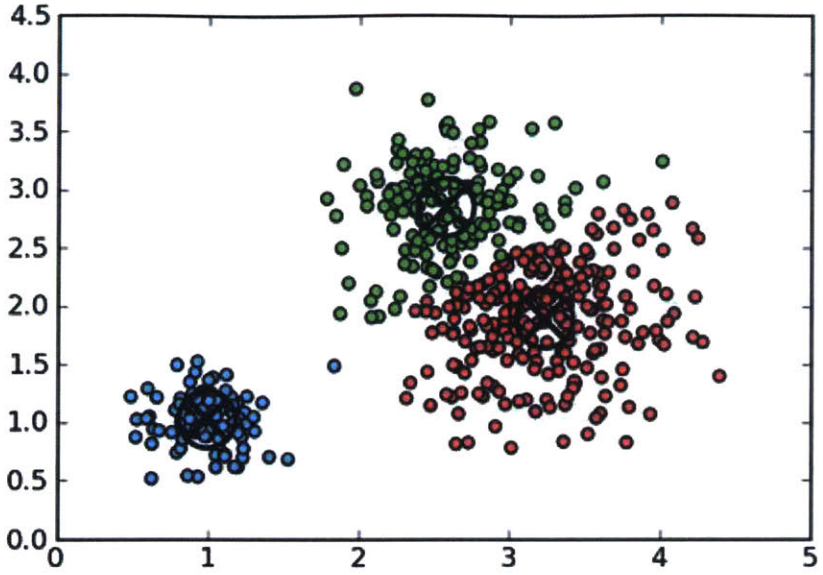


Figure 5.10. Observations separated into three classes.

Now that each image belongs to a cluster, we can compute the training data for the SVM classifiers. Each training image undergoes SIFT computing to obtain descriptors and then associates the descriptors with a cluster. The histogram of similar features can then be used to train an SVM to classify unseen images (Figure 5.11).

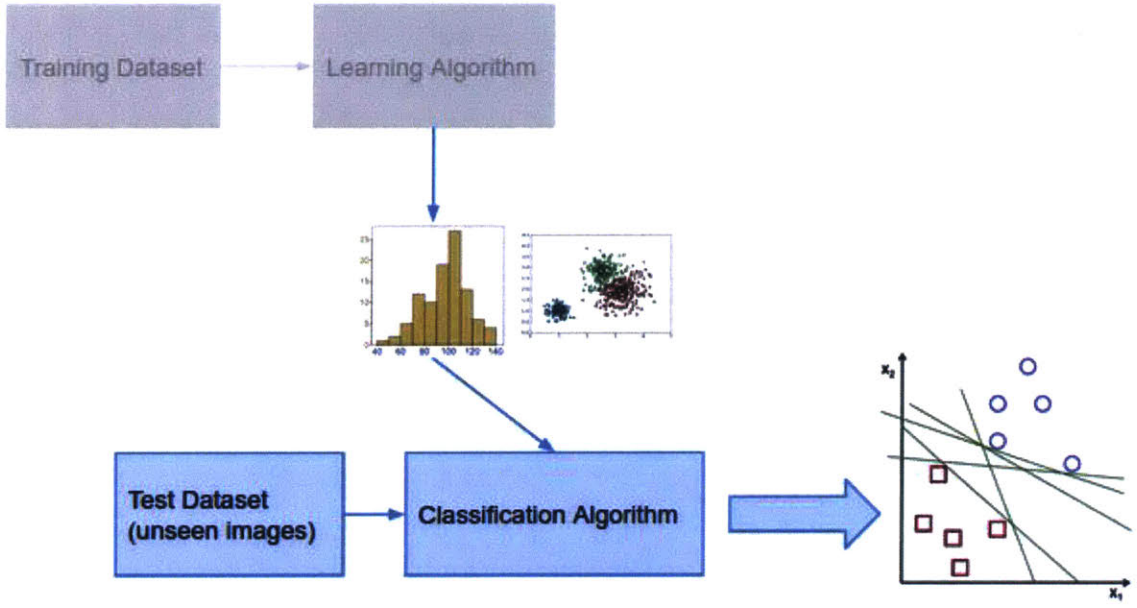


Figure 5.11 Steps for classification

Now that images are sorted, we can begin to extract the camera poses from the desired 2-dimensional images. To do so, understanding how the camera model in computer graphics work is essential to accomplishing such a task.

5.2.5 Camera Model

The simplest camera is the pinhole camera and is the model used in computer graphics. A pinhole camera is an imaging device that is sometimes the size of a small box or an enclosed room with a small hole on one side. Light enters through the opening and projects an inverted image on the opposite side of the chamber (Figure 5.12). In this model, an image plane is set

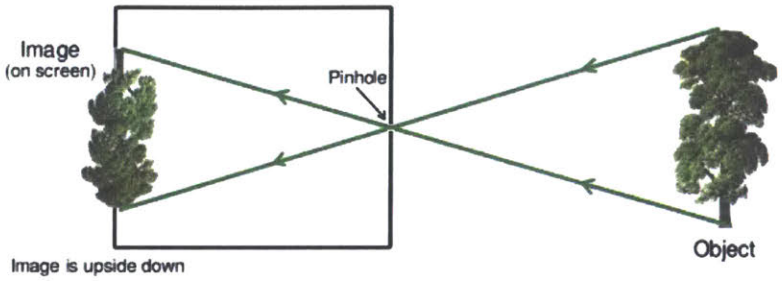


Figure 5.12 The geometry of a pinhole camera

some distance from the camera center and a point is mapped into the image plane through a straight line that also originates from the camera center (Figure 5.13). The camera model can be described as a matrix containing 1) intrinsic parameters that include the focal length and distortion, and 2) extrinsic parameters that includes rotation and translation.

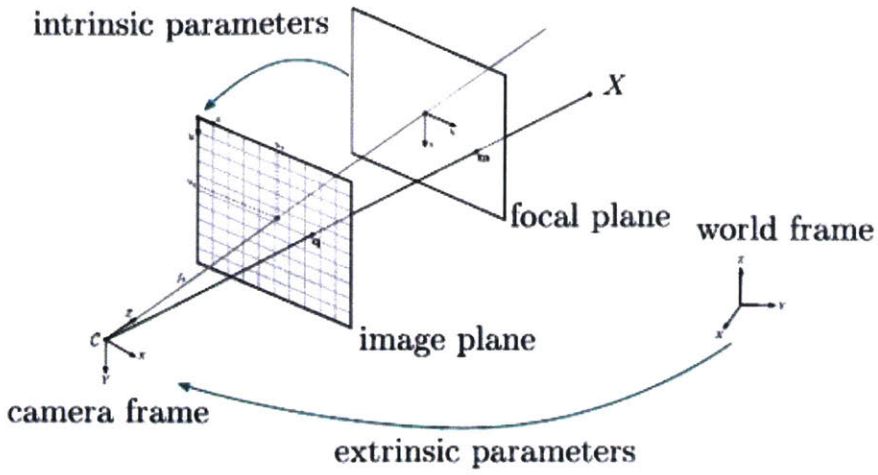


Figure 5.13 A typical pinhole camera model for computer graphics

The camera model can be decomposed into extrinsic and intrinsic parameters. Described in matrix form, the camera matrix contains two matrices: intrinsic matrix, K, and extrinsic matrix, $[R \mid -RC]$:

$$P = K[R \mid -RC] \qquad \textbf{(Eq. 5.5)}$$

K is a 3x3 matrix describing the camera's internal parameters like focal length and radial distortion, R is rotation, and vector C is the camera center. From the camera model, the extrinsic camera matrix takes the form of a 3x3 rotation matrix and a 3x1 translation column-vector as in Equation 5.6, R is the rotation and t is the translation:

$$P = [R|t] = \begin{bmatrix} r_1 & r_2 & r_3 & t_1 \\ r_4 & r_5 & r_6 & t_2 \\ r_7 & r_8 & r_9 & t_3 \end{bmatrix}$$

Eq 5.6 Extrinsic camera matrix

The vector **t** is the position of the world origin in camera coordinates and r is the direction of the world axes. It's important to note that the extrinsic matrix describes how the world is transformed relative to the camera. Finding the 3D position of a camera is found by Equation 5.7.

$$-R' * t \qquad \textbf{(Eq. 5.7)}$$

The process of estimating camera positions and three-dimensional structure from a sequence of two-dimensional images relies on the camera model within a technique called, Structure from Motion (SfM). The method was effectively developed in the late 1980s, but recent photogrammetry techniques such as bundle adjustment is now regarded as the "gold standard for performing optimal 3D reconstructions from correspondences[21]." The next section describes Structure from Motion concepts.

5.2.6 Structure from Motion

Several industries such as manufacturing, entertainment, architecture/engineering/construction, and archaeology, for example, perform work in virtual environments with 3D models. User driven 3D modeling, however, is prone to errors and can become a lengthy process that can increase costs. The availability of high-definition laser scanning technology exists, yet the cost of owning and caring for such equipment has not yet decreased for casual consumer handling. The cost-effective alternative is the use of image sequences from consumer cameras, which are cheap, lightweight, and precise to several megapixels. An example of Structure from Motion in action was performed by a research team of students (myself included) with Professors Takehiko Nagakura of MIT, visiting professor Howard Burns, and MIT Research Fellow Daniel Tsai. In this research, we piloted a custom drone to circumvent historic Palladian Villas within the vicinity of Vicenza, Italy. Each image sequence was then processed for 3D reconstruction

[21] Snavely, Noah, Steven Seitz, and Richard Szeliski. "Modeling the World from Internet Photo Collections." International Journal of Computer Vision. Volume 80 Issue 2, Nov. 2008. Pages 189-210. http://phototour.cs.washington.edu/ModelingTheWorld_ijcv07.pdf

using open source software tools to produce the following textured mesh[22] resulting in Figure 5.14.

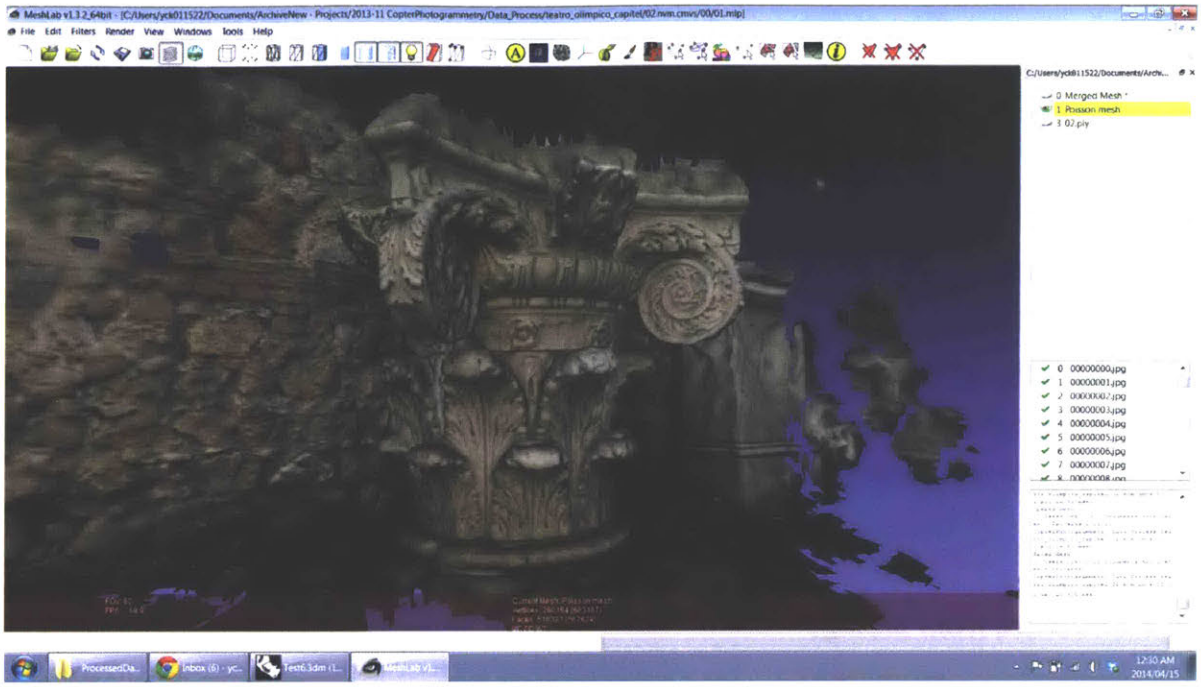


Figure 5.14 High polygon mesh with 260,154 vertices of a Corinthian column at the Teatro Olimpico in Vicenza, Italy.

With enough images whose viewpoint overlap one another, the reconstruction process can produce details to create a 3D object for reuse in virtual environments. The edges appear jagged due to the lack of image registration points or non-matching features found by the image matching algorithm. To create a 3D model from a sequence of images, the image dataset must be devoid of "incorrect" images so that multiple pairs of similar images can be compared against one another. Creating a 3D point then relies on the left and right image pair in a sequence (Figure 5.15).

[22] Nagakura, Takehiko, Daniel Tsai, and Joshua Choi. "Capturing History Bit by Bit - Architectural Database of Photogrammetric Model and Panoramic Video." Proceedings of the 33rd eCAADe Conference - Volume 1, Vienna University of Technology, Vienna, Austria, 16-18 September 2015, pp. 685-694. http://papers.cumincad.org/cgi-bin/works/BrowseAZ&name=authors/Show?ecaade2015_110

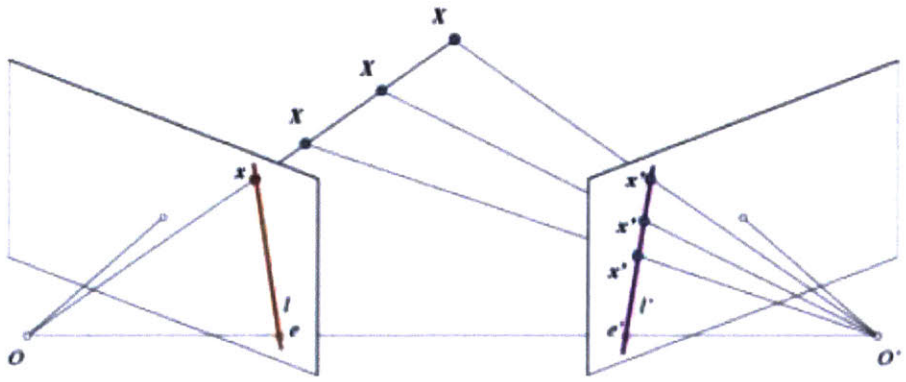


Figure 5.15 Comparison of points (*x* and *x'*) between the left and right image to estimate *X*, the 3D point

The points on the left image (Ox) and in the right image (O'x') of Figure 5.15 are used to triangulate the correct 3d point. This geometry is encapsulated by the fundamental matrix, **F**, a 3x3 matrix of rank 2. In computer vision, the fundamental matrix corresponds points, ***x*** and ***x'*** *where* ***Fx*** is the line, or epipolar line, that describes the corresponding points between images -- this can occur even if the inner camera parameters, like focal length, are unknown. The corresponding image points show a relation as $x'^{T}Fx = 0$. If the inner camera parameters are known, then the essential matrix may be used for point to point correspondence between images as well as to determine the relative position and orientation between cameras and 3D position of image points.

The essential matrix is a 3x3 matrix described as:

$$E = R[t]_x \qquad \textbf{(Eq.5.6)}$$

R is a 3x3 rotation matrix and **t** is a 3-dimensional translation vector and $[t]_x$ represents the matrix cross product with **t**.

Extracting the essential and fundamental parameters from images can be extracted through Bundler, an open source tool that runs from a terminal's shell script for building structure-from-motion 3D reconstructions, the systems was built by Noah Snavely at the University of Washington that analyzes unordered image collections. Bundler extracts keypoint features using SIFT to match features between a set of images and outputs model and texture coordinates, as well as extrinsic and intrinsic camera parameters to reconstruct a scene.

While Bundler has traditionally been used for reconstructing 3d scenes, or sparse point clouds, from a set of images, the camera estimation parameters that the system yields can be used for the purpose of calculating the camera pose of each image. Bundler outputs two very useful mathematical objects for estimating camera objects: external and intrinsic camera parameters. Both the external and intrinsic camera parameters are encapsulated by the fundamental matrix, whereas the essential matrix only encapsulates extrinsic parameters.

Bundler requires accurate information about the intrinsic parameters for each photograph in the collection, such as the focal length found in an image's metadata or EXIF tag in order to perform the SIFT algorithm. An essential parameter in the EXIF file is the focal length and CCD width, a camera's image sensor. What happens when images do not contain the original EXIF tags? Bundler relies on a CCD sensor look up table if an image does not contain that information in its EXIF tag, however, should an image be taken with a newer camera, it is unlikely that Bundler contains the necessary information to SIFT images. In that case, the user must input the ccd width information into the CCD database. If no information is given, Bundler omits the image from calculation. VisualSFM, a closed-sourced SfM GUI software, created in 2006 and led by ChangChang Wu of the University of washington GRAIL lab, automatically calculates focal lengths even if the original EXIF does not contain the focal length parameter. VisualSFM sets the focal lengths to be 1.2 * max(width, height) of the image, corresponding to a medium viewing angle.

So far, this exploration uses Flickr to obtain images of buildings partly because the original EXIF tags remain in the image - with a couple of scripts to obtain the information. The vast usage of Facebook and Instagram for user uploaded images, however, trumps Flickr in terms of uploaded content. A large user base of up-to-date streaming images can provide valuable information within minutes of posting instead of days as seen in Flickr. Unlike Flickr, Facebook

and Instagram automatically strip away EXIF tag information for privacy reasons. In such a case, Bundler cannot handle the SIFT process without the EXIF data while VisualSFM can calculate the parameters necessary to carry through with the SIFT algorithm.

A typical use case for Bundler comes after several custom scripts have scraped and cleaned the image dataset, this will be discussed in more detail in the section. The general flow for using Bundler:

1. Scrape images from Flickr and output urls, tags, save and create downloaded images to directory. Save urls and tags in a new text file.
2. Extract raw url output.
3. Run regular expression to extract only urls in text.
4. Get original size images with Flickr IDs, extracted from urls in step 3.
5. Resize images.
6. Prepare and instantiate Bundler for image processing.

I have developed an initial test script within Blender, an open source 3D modeling software, to calculate the camera and image positions and (faint appearing) 3D point cloud from the intrinsic and extrinsic parameters outputted from Bundler and VisualSFM in Figure 5.16. While the test script was useful for calculating the camera positions with existing camera and surface objects from Blender, the software was not the tool of choice for viewing the point cloud or camera objects for data analysis and specific visualization objectives. In the next chapter, I discuss the major step along the path towards data analysis and custom visualization: data mining and cleansing.

Figure 5.16 Test script with images and cameras positions obtained from 2D photo collections.

Chapter 6

Taming the Wild: Methods in Data Mining

6.1 Introduction

With the explosion of massive, high-dimensional, distributed datasets in the last decade, data mining research continues to invent new algorithms, systems, and infrastructure to accommodate data driven applications[23]. Data mining itself is a multidisciplinary field that includes statistics, machine learning, database technology, data visualization, pattern recognition, information theory, and artificial intelligence. Data mining differs from conventional statistics in that statistics is based on a human intervened form of hypothesis creation and validation testing, whereas data mining is a semiautomatic process of learning from data (usually large datasets) --- extracting rules through pattern recognition, developing predictive models, and employing methods of supervised and unsupervised learning.

This research encompasses both supervised and unsupervised learning problems. A supervised learning model takes a human-guided training set, inputs the set into an algorithm that builds a prediction model from observing the input, and predicts the outcome of similar but unseen information. For example, as you'll see later in this chapter, a supervised learning model was used for classifying images of buildings: similar buildings are grouped together and

[23] Sumathi, S. and Sivanandam, S.N. Introduction to Data Mining and its Applications (Studies in Computational Intelligence). 1st edn. Springer. 2006.

human confirmed for correctness before the set is fed into the classification algorithm. When the classification algorithm encounters an unseen image not existent in its training set, a probabilistic model is used for classifying what category the image, or building, belongs to. The opposite of a supervised model is an unsupervised one. Going back to the image example, instead of having a human guide the training set, the algorithm tries to sort the images on its own by evaluating and comparing each image's nearest neighbor.

Due to the large size datasets in this research, arriving at the knowledge discovery stage can be hampered by the lack of computational power from a single machine. The amount of computation involved sometimes require parallel processing as an essential component. I'll discuss the experimentation and techniques using open source cluster-computing frameworks later in this chapter.

Processing and cleaning raw datasets consumes the most time in any data analytics cycle, the same is true for this exploration. The variety of data types explored - images, WiFi, and natural language text, requires a unique approach in order to create useable information so that any of the algorithms in computer vision, machine learning, and analytics can be consumed reliably. This chapter explains the process, strategies, and trials and tribulations for handling different types of information.

In a traditional structure from motion pipeline, for instance, the person creating the 3D model from images is also the person who takes the pictures from a single camera. The images taken contain the original metadata such as focal length, date and time, camera sensor width, and other necessary information that a structure from motion algorithm relies upon during the reconstruction process. Internet photos, on the other hand, especially when uploaded to social networking sites such as Facebook, Twitter, and Flickr, are automatically stripped from the original metadata for the purposes of user privacy. What is left in the metadata are image size and the photograph's file name. Not very useful for an SfM pipeline. This chapter discusses how to overcome that obstacle.

As for the MIT WiFi dataset, disparate pieces of data require extraction from different databases and ultimately merged to create meaningful information. The standalone WiFi logs contain table columns for the number of devices connected to a WiFi access point ID, but without knowing

where the WiFi AP is located on campus, the logs by themselves narrow the scope for data analysis. In this case, a separate database query must be entered to retrieve the WiFi IDs along with their building and room numbers. Combining WiFi locations with the log data provides a bigger picture of how people move around campus.

Lastly, data mining natural language text from image captions also requires a scrubbing process. Raw tweets contain punctuation marks, capital letters, and sometimes a wild ASCII ensemble that any data analysis algorithm cannot interpret. Removing all of the above including "stop words," or common natural language words such as "the", "is", "a", is a typical process that accompanies text mining.

6.2 Data Mining Workflow

Each of the steps to acquire and reproduce information in a format that fits the analysis objectives share similar workflows. The data mining workflow contains multiple steps such as raw data collection, cleansing, and processing. Figure 6.1 shows the data processing pipeline where the typical workflow contains the following phases:

1. *Raw data collection:* Different types of data require a unique approach for extraction and is typically application specific. This stage proves critically important because it can impact the data processing workflow further down the pipeline. At this stage, the raw collection is placed in a database, or data warehouse for processing.

2. *Data cleansing:* The data collected is often in an unsuitable format and unfriendly to data mining algorithms.[24] In order to make the data suitable for processing and analysis, extracting a relevant set of features or attributes becomes a necessary step. In many cases, the data contains missing information and/or erroneous parts where an estimation or correction must take place.

3. *Algorithms and analysis:* This stage is to design methods for analyzing the processed data, it often involves selecting and/or designing algorithms that best fit the data.

[24] Baughman, A., Gao, J. et.all (Eds.). Multimedia Data Mining and Analytics, Springer. 2015.

Designing the methods ultimately depends upon the objectives and formulated questions about the dataset at hand.

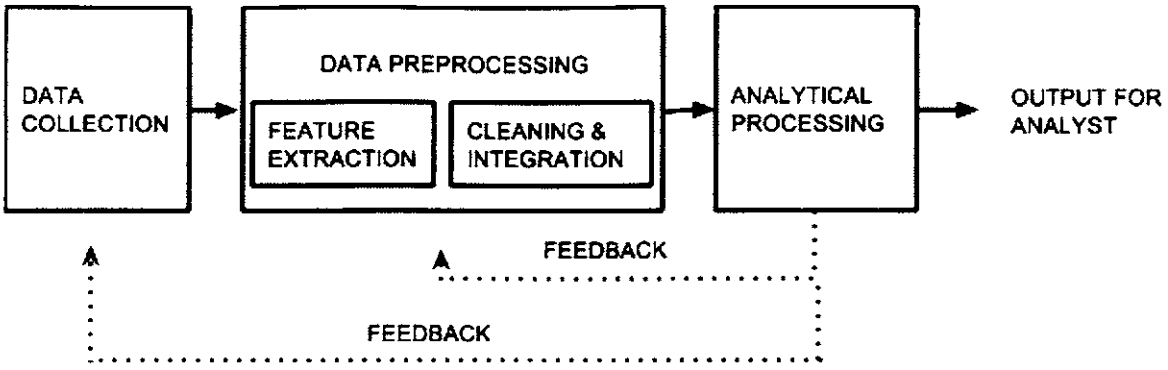


Figure 6.1 The data processing pipeline (adapted from [2]).

6.2.1 Content-based Image Search and Retrieval (with Flickr)

With the data mining workflow in mind, the process of collecting thousands of public images from Flickr, a user-uploaded photo sharing website created in 2004, follows a similar process with the following details in Figure 6.2 and pseudocode for querying images.

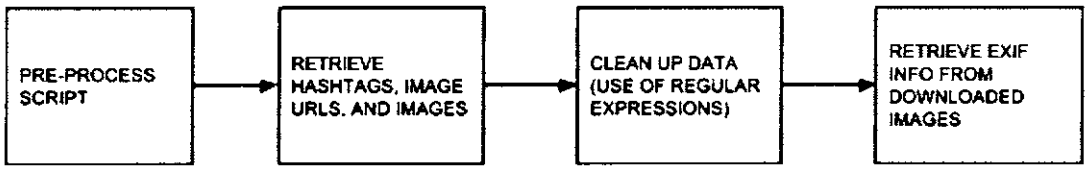


Figure 6.2 Image retrieval and clean up process pipeline.

Algorithm 6.1. Scraping Flickr images

```
Insert k number of items to retrieve.
Open text file with query words.

for query_tag in query file:
     crawl Flickr by tag, url
     for photo in photos:
          set sleep timer
          if url is not None:
               print query tag, current index, url
               retrieve url path
               download photos to current path
               if current index >= number of items toretrieve:
                    break
               else:
                   print 'no url'
```

In this case, a series of scripts have been created to extract the images by first inputting the following parameters: keyword search, number of images to collect, outdoor/indoor, and/or geoboundary. In this case, the keywords tested include, "statacenter", "mitchapel", "carpentercenterforthevisualarts", and "mitmedialab." The script returns the keyword search, image url, and accompanying hashtags for each image (Figure 6.3). The hashtags serve as useful information for text analysis later. Downloading the images require extracting each image's url from the script's output.

```
statacenter
1
statacenter 0 https://farm4.staticflickr.com/3943/15386021170_c2607cb639_z.jpg
barcelona from blue cambridge arizona building strange architecture modern store interesting
spain funny downtown distorted tucson contemporary massachusetts dramatic surreal simulation
explore architect gaudi novel hmmm miles unusual frankgehry statacenter department whimsical
dotcom imitation aliceinwonderland usbank thousands mits childlike influence simulated
antonigaudi chileverde deconstructionist inventive affecting architecturalfirm imitated
explored lerners makesyouwonder architecturaleffect thegehryeffect gehryeffectbuilding
gaudiinbarcelonaspain gaudisinfluenceongehry onecongressstreet seaverfranksarchitects
statacenter 1 https://farm4.staticflickr.com/3930/15234778198_8ee005f020_z.jpg
cambridge boston mit statacenter
statacenter 2 https://farm4.staticflickr.com/3928/15398312856_ec5d3951e3_z.jpg
cambridge boston mit statacenter
statacenter 3 https://farm4.staticflickr.com/3929/15234681840_79ce678526_z.jpg
cambridge boston mit statacenter
statacenter 4 https://farm3.staticflickr.com/2948/15418186381_b441f682f1_z.jpg
cambridge boston mit statacenter
statacenter 5 https://farm4.staticflickr.com/3929/15388948206_3c38b49c35_z.jpg
cambridge boston mit statacenter
statacenter 6 https://farm4.staticflickr.com/3929/15411992955_b9d18756d5_z.jpg
cambridge boston mit statacenter
```

Figure 6.3 Initial image scraping process of 500+ images of Stata Center on Flickr.

The goals for gathering multiple images of Stata Center involves the following:

1. Create an image-based 3D model through image reconstruction.
2. Construct the camera pose, rotation and translation, for each image.

One of the caveats in data mining pictures of a building from Flickr is that the results returned also contain images of people and activities happening inside of the building. The results returned about Stata Center not only include the targeted images of Stata Center's exterior facade, but also images of people and things associated with the events that occur inside of Stata Center such as Tim the Beaver during an opening ceremony of Charles Vest Street, or an MIT hack memorabilia of a police car that famously sits in an indoor loft, Figure 6.4. Images of people and activities can reveal how individuals use public spaces as a separate data mining objective, however, incorporating images of people (or Tim the Beaver) as part of the dataset yields an inaccurate image-based 3D model.

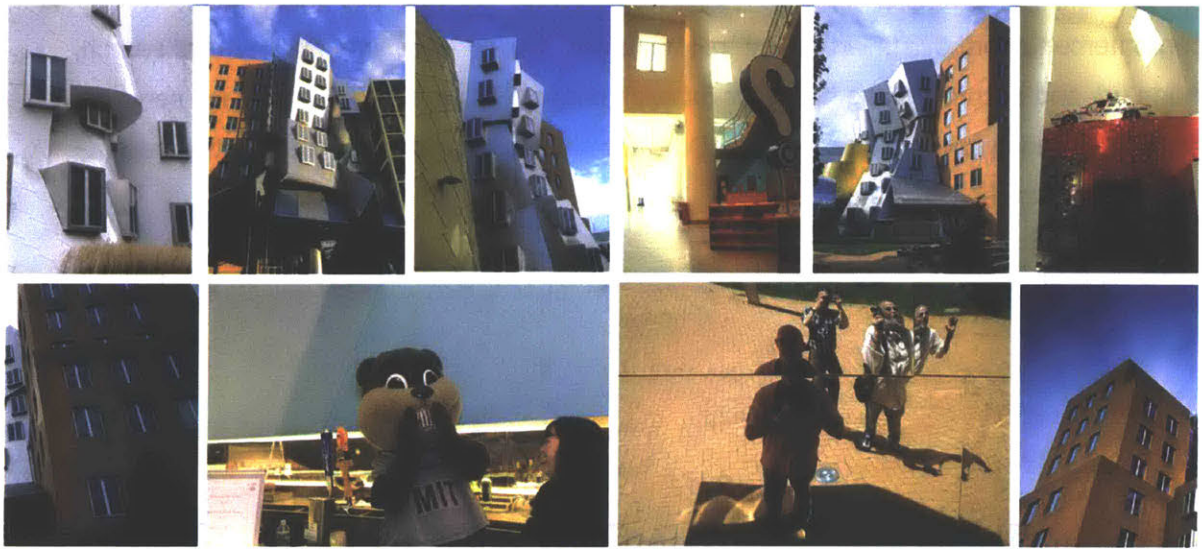


Figure 6.4 The targeted images are exterior shots of the building, yet images of people and interior shots are amongst the collection returned.

6.3 Wrangling with Noisy Data

Over 3,000 images are returned from scraping Flickr -- hand sorting the images can prove less than ideal. To create the image based 3D model, we need to use images that are useful for the image reconstruction algorithm in order to produce accurate camera extrinsics. For example, leaving the images that contain self-portraits taken at Stata Center can yield erroneous results when trying to match similar images together (Figure 6.5), notice how the SIFT computer vision algorithm somehow finds structural similarities between the woman's facial features and the protruded window edges of Stata Center. To avoid matching errors, the aid of machine learning algorithms help sort images by similarity and employing techniques of supervised learning as explained later in this section. Automatically sorting images requires an algorithm to extract unique features from each image, or feature extraction. Image data is represented as pixels where color, gradient, and edges can begin to form the basis for a semantically rich representation of an image and be used for comparison against other images.

Depending on the building and its popularity, if the images are too different from one another such that a feature point in one image cannot be found in another, then an epipolar line cannot be drawn or yields incorrect results. The diverse set of images can prove a challenging task when a sparse images do not yield enough results to create a motion path to connect the images. When the path cannot be derived, the next approach is to increase the image retrieval size from (i.e. from 500 to 1000 or more). Increasing the amount of images also increases the time to compute the Structure from Motion process. Using large sized images does not equate to dense point clouds, it just unnecessarily increases the computation time of the Structure from Motion process.

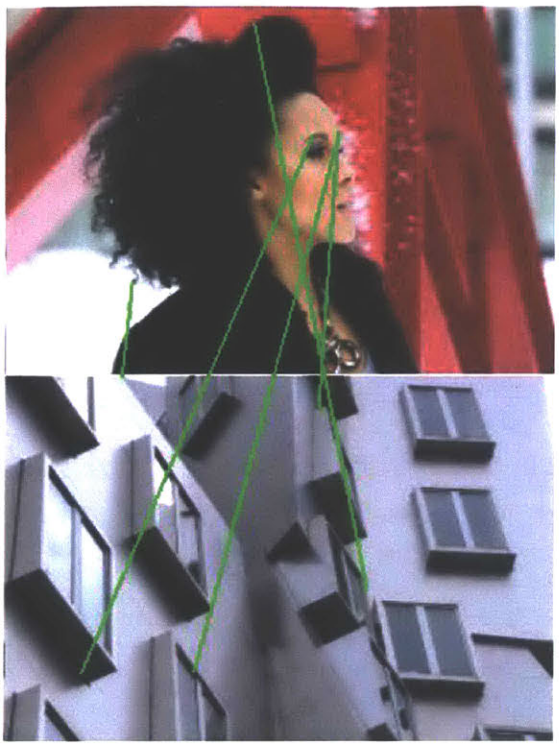


Figure 6.5 A mismatch from the SIFT algorithm

6.3.1 Data Classification

The classification algorithm is the ability to learn by example so as to predict which category a data sample belongs to. In this case, the classifier correctly predicts and identifies whether an unseen image belongs to a particular category. That is, all buildings of "stata center" and "baker house" are automatically classified and sorted to each its own class. Further classification labels can occur at the level of outdoor vs. indoor images or people vs. things.

The algorithm requires a training input, or a ground truth dataset, typically handpicked as the accurate collection of information. The training data, denoted by D contains n data points and

d features. Each training data point D is associated with a class label from $\{1...k\}$. Features, d, teaches the algorithm multiple possible input conditions. As an example, features can be SIFT descriptors or keys for an image. Recall that the SIFT keys, or feature vectors, is based on image measurements that create particular interest points. Figure 6.6 illustrates the use of SIFT to extract feature keypoints in an image of Stata Center. Each keypoint detection contains unique attributes such as its (x,y) coordinates, angle of orientation, and size of neighborhood.

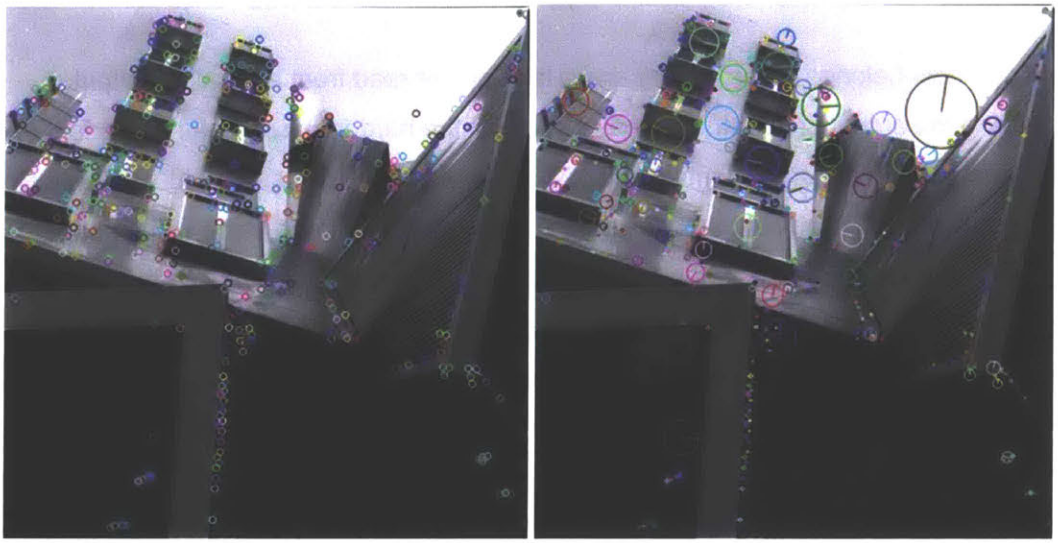


Figure 6.6 *Left.* Finds the keypoint in the image. *Right.* Keypoints highlighted as circles along with its orientation.

Class labels accompany features as a correct map of inputs to output. For example, in the case of 3 classes/categories of buildings: Stata Center, Baker House, and the Carpenter Center at Harvard. A label for each class can look like Table 6.1:

Category	Label
Stata Center	0
Baker House	1
Carpenter Center	2

Table 6.1 Numeric labels assigned to each class.

A test script outputs codified labels to determine which class an unseen image belongs to when using the classifier algorithm. A cross validation of the training phase outputs a cross validation rate of 82.4561% , c=128.0, g=0.0001220703125 where c is the penalty parameter and g, gamma, is the parameter for the SVM margin's cost function which controls each support vector (for stability). Classifying an image loads feature vectors, computes the visual words histogram, writes the histogram to file and passes it to the SVM algorithm, then cross validates the dataset. The output prediction is either 0, 1, or 2.

Each image now belongs to a category and a labeled set read from a terminal output. Materializing the output so that the images stored on the harddrive can belong to a folder with label categories (i.e. folder "0" contains all of the images of Stata Center, folder "1" with Baker House, and so forth) is the next step towards easing the transition towards using image data for visualization and analysis. In other words, The classifier algorithm assigns a label for each image so as to indicate which class an image belongs to (i.e. Stata = 0; Baker = 1). So far, the algorithm outputs the result in a terminal console. To associate each image and class into sets requires a process of further automation -- a script runs the classifier on every image located in the test folder and then another automation script prepares the results for further cleaning and processing. Algorithm 6.1 in pseudocode walks through the process of extracting the images and labels from the SVM output so as to group same set images into a corresponding label directory.

Algorithm 6.1: Sort Classified Images into Bins

```
Input: input digital image of M x N pixels.
Output: A = {a: 1.jpg, 2.jpg,...n.jpg}
        B = {b: 5.jpg, 10.jpg, 8.jpg...n.jpg}..

function SortClassifiedImagesIntoBins:
      Read image of Mx N pixels, i, in directory
          for every image in directory do
              Compute SIFT
              Compute k-means clustering
              Compute SVM for Classification

     #With regular expressions, clean up and extract labels from
      classification
      open "LabelPredictions.txt"
      findall and match strings ".jpg" and label values 0..n
      a = list of classified values
      b = list of images in folder
      Create set (a,b)
       for each set
          extract images that belong to label 0,1,2,...n.
          Get keys in dictionary
          Use keys to create directory
          Copy files to separate label folder in directory
    OUTPUT[(a: 1.jpg,2.jpg,3.jpg..n.jpg), (b: 5.jpg, 10.jpg, 8.jpg..n.jpg)]
```

Grouping similarly predicted images makes it easier to feed the reconstruction algorithm an image set (Figure 6.7), otherwise, feeding the reconstruction algorithm a random set would not compute matches between images in order to triangulate similar points.

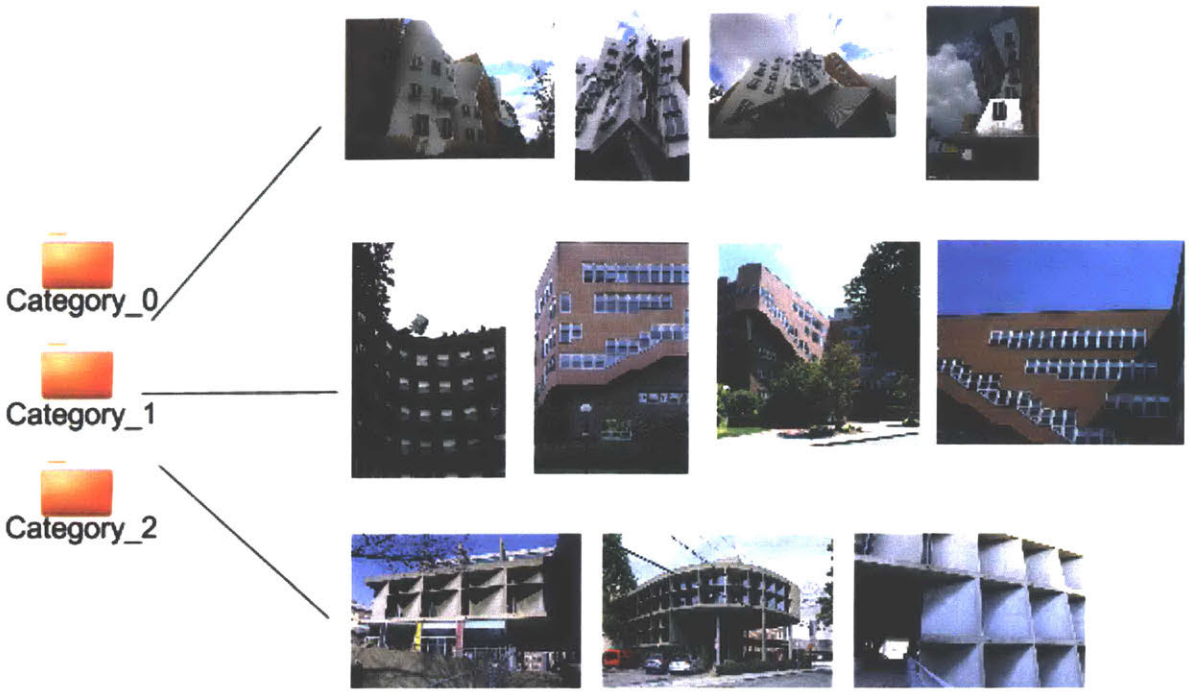


Figure 6.7 Output of Algorithm 6.1 on a set of test images where the algorithm creates several directories with similarly grouped images by class labels. Category 0: Stata Center; Category 1: Baker House; Category 2: Harvard Carpenter Center

6.3.2 Camera Pose Data Cleansing and Extraction

After estimating the camera position for each image, taking the output file requires another data cleansing and extraction process to ultimately create a useable format for building a web application. The structure-from-motion output serves as the input for the clustering image algorithm, Clusterings Views for Multi-view Stereo (CMVS), which in turn outputs an N-View match file. The N-View match file contains input image paths and multiple 3D models. Each reconstructed model contains the following:

```
<Number of cameras>   <List of cameras>
<Number of 3D points> <List of points>
```

The cameras and 3D points contains the following format:

```
<Camera> = <File name> <focal length> <quaternion WXYZ> <camera center><radial
distortion> 0
<Point>  = <XYZ> <RGB> <number of measurements> <List of Measurements>
<Measurement> = <Image index> <Feature Index> <xy>
```

The camera model in Table 6.2, for example, is clustered as a set of five similar feature points. These are the necessary parameters for placing the camera objects and their orientation in the 3D visualization software. The nvm outputs the camera rotation as a quaternion, or a 4D rotation vector as $[w, (x,y,z)]$ that has four real numbers - three is the x,y,z vertex that forms the direction, and the fourth number represents a rotation around its own axis. Thus, a quaternion stores a rotation axis and a rotation angle (Figure 6.8).[25]

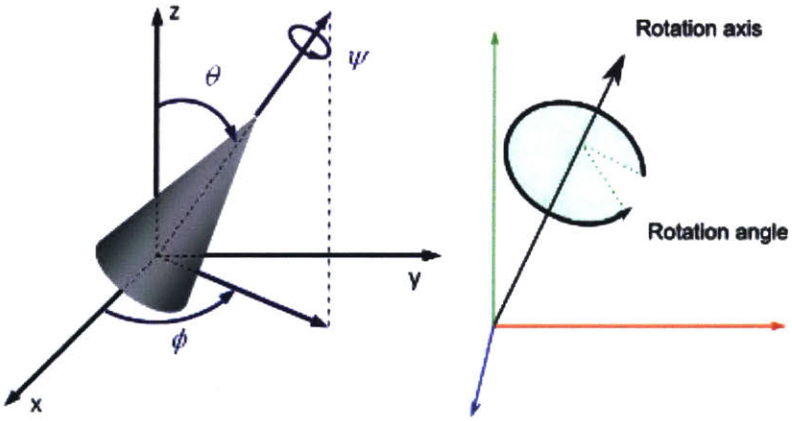


Figure 6.8 A three-dimensional rigid body rotation.

To generate the camera orientation, the quaternion needs to convert to a 3x3 rotation matrix (for ease of use), which is a basic rotation in $\mathbb{R}^3$ coordinate system. Each row of the rotation matrix

[25] Bomfim, Diney. "Cameras On Opengl ES 2.X - The ModelViewProjection Matrix".*Blog.db-in.com*. N. p., 2014.. 22 Sept. 2016. http://blog.db-in.com/cameras-on-opengl-es-2-x/

gives one axis as in Equation 6.1 where Rx, Ry, and Rz denote rotation about the x, y, z axis. Converting from quaternion to a rotation matrix is defined in Equation 6.2[26] [27].

$$R_X(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & Cos(\theta) & -Sin(\theta) \\ 0 & Sin(\theta) & Cos(\theta) \end{bmatrix}$$

$$R_Y(\theta) = \begin{bmatrix} Cos(\theta) & 0 & -Sin(\theta) \\ 0 & 1 & 0 \\ Sin(\theta) & 0 & Cos(\theta) \end{bmatrix}$$

$$R_Z(\theta) = \begin{bmatrix} Cos(\theta) & Sin(\theta) & 0 \\ -Sin(\theta) & Cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Equation 6.1 Three basic rotation matrices that rotate vectors by an angle θ about the x, y, or z axis.

$$\begin{bmatrix} 1-2y^2-2z^2 & 2xy-2wz & 2xz+2wy \\ 2xy+2wz & 1-2x^2-2z^2 & 2yz-2wx \\ 2xz-2wy & 2yz+2wx & 1-2x^2-2y^2 \end{bmatrix}$$

Equation 6.2 Quaternion to matrix

Essentially, the NVM output is a tab-delimited file whose columns require extraction and conversion to a 3x3 matrix.

[26] Goldstein, Herbert, Charles P. Poole Jr., et al. Classical Mechanics (3rd edition). Essex: Pearson, 2014.

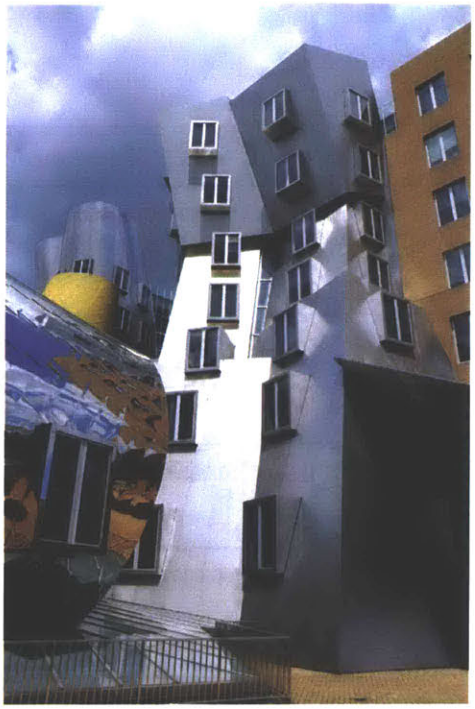
[27] Arfken, George B., Hans J. Weber, et al. Mathematical Models for Physicists: A Comprehensive Guide. 7th edition. Boston: Elsevier, 2013.

File name	Focal Length	Quaternion (wxyz)	Camera Center	Radial Distortion
..\stata150.jpg	5133.75683594	0.992323803901, -0.0552800499587, 0.110512261949, -0.00495539072888	-0.00111131045375, -0.000281950209713, 0.00220157750599	0.0394807194039
..\stata161.jpg	6440.20947266	0.894099603329, -0.0789708419637, -0.417834373492, -0.140584146311	-0.963669225534, -0.0892060724903, 0.240188348499	-0.133304967428
..\stata133.jpg	2280.24438477	0.999999956616, 0, 0, 0	-0.164572413731, -0.042303181876, 0.289858485042	0.00538660032483
..\stata147.jpg	3126.55883789	0.995096038635, -0.0164107107858, 0.0973293392044, 0.00644124667343	-0.0593965394761, -0.0479269062805, 0.191456089363	-0.000286548274023
..\stata126.jpg	2502.24536133	0.948645638937, -0.00799275146411, -0.287701732216, -0.131283155476,	-0.720614778297, -0.103371781126, 0.30048788878	-0.014604388447

Table 6.2 Decomposing an n-view match output.

6.4 Camera Settings and Scene Information from Images

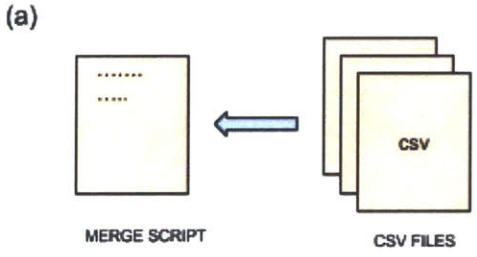
While each camera reveals an interesting set of clusters and rotational positions, their associated images contain an essential ingredient for tying in external datasets that may shed further insights into the analysis, namely, the "time" variable. All of the images downloaded from Flickr are original content that contain meta information, or EXIF (Exchangeable Image File) data. EXIF information records a camera's settings such as shutter speed, date and time a picture was taken, exposure, focal length, metering pattern, sometimes GPS coordinates, and others (see Appendix 2 to view other fields associated with an image). Figure 6.9 shows truncated EXIF information adjacent to its accompanying image.



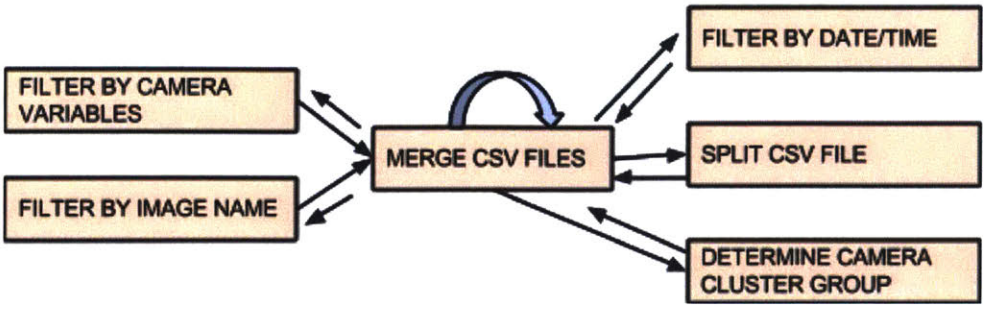
File name: stata99.jpg
File size: 3756049 bytes
File date: 2015:05:02 13:15:21
Camera make : FUJIFILM
Camera model: X-E1
Date/Time: 2014:05:18 09:30:36
Resolution: 3264 x 4896
Flash used: No
Focal length: 18.0mm (35mm equivalent: 27mm)
CCD width: 23.40mm
Exposure time: 0.0005 s (1/1900)
Aperture: f/6.4
ISO equiv. : 400
Exposure bias: 0.33
Whitebalance : Auto
Metering Mode: pattern
Exposure: aperture priority (semi-auto)
JPEG Quality: 97

Figure 6.9 A user uploaded image and its meta information.

Reading, writing, and editing image meta data is a trivial process with an open source command line tool like ExifTool by Phil Harvey. For this research purpose, every image's filename and date/time fields were extracted and merged with its camera pose variable to finally arrive at a useable (clean) data file for analysis and visualization shown in Figure 6.10 as a csv (comma separated value) formatted file.



(b)



(c)

```
"id","FileName","Date","camVariables","imgObjVariables"
"1","stata1313.jpg","2010:10:23 10:32:36","camera_pose1","imageObject1"
"2","stata1.jpg","2013:09:12 16:18:34","camera_pose2","imageObject2"
"3","stata254.jpg","2007:11:24 14:51:19","camera_pose3","imageObject3"
"4","stata1113.jpg","2011:01:05 09:11:18","camera_pose4","imageObject4"
"5","stata1033.jpg","2008:07:20 12:19:23","camera_pose5","imageObject5"
```

Figure 6.10 (a), (b). A merge script splits and extracts the necessary fields from each csv file to create a "clean" data file, (c). The "camera_pose" variable represents a 3D camera object that holds the image's camera matrix; "image_object" refers to the associated surface mesh for displaying the image in the web viewer.

It's important to note that in most cases for any data scrubbing activity, missing values and outliers are expected. This particular process was no exception, some images omit the date/time fields in the EXIF header. Despite the missing values, the images and camera poses without a time component remain as static 3D objects in the visual application. Dealing with missing values and outliers is at the discretion of the analyst, the final presentation of results derive from a handpicked dataset. As described in chapters 3 and 4, the results of the data mining and cleansing process facilitates the discovery of relationships between EXIF metadata and other data attributes for the Stata Center, Simmons Hall, and Copley Square image datasets.

6.5 Exploring WiFi Datasets (4,762 WiFi Access Points)

Just as raw data from images require a clean up phase, the same holds true for the raw MIT WiFi dataset provided by MIT CSAIL and IS&T. The WiFi logs collected range from September 2014 to April 2015 with some gaps due to IS&T service shut downs. The raw data log (Table 6.3) contains a timestamp, number of connected devices to a WiFi access point, and a WiFi access point ID. Combining the dataset with a base map of MIT buildings and its latitude and longitude helps visualize how time and the density of connected devices around MIT relate to one another.

Timestamp	No. of Connected Devices	WiFi Access Point ID
1411676653	0	0
1411676653	4	1
1411676653	6	2
1412401753	3	4739
1412401753	1	4753
1412401753	1	4195

Table 6.3 MIT WiFi Access Log

To do so, generating scripts to scrape the latitude and longitude of buildings and MIT building geometry and their names (i.e. building N51) becomes possible through MIT's map REST API service (Figure 6.11 and Algorithm 6.2).

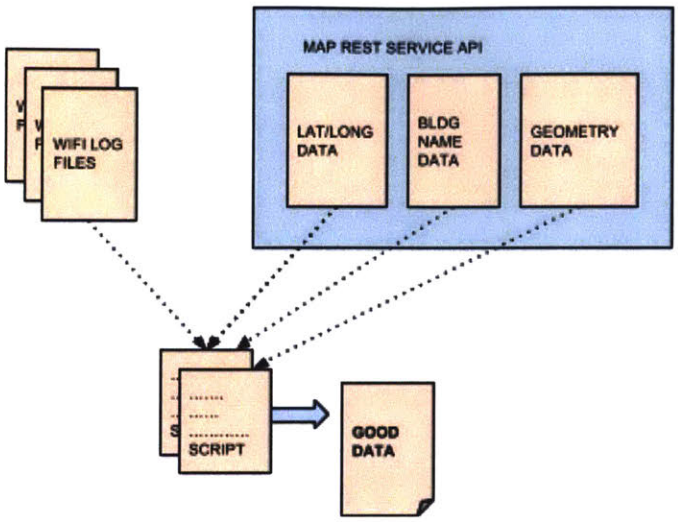


Figure 6.11 Obtaining a usable dataset requires the creation of scripts that scrape and merge separate fields.

Algorithm 6.2. Scrape & Merge WiFi Data

```
input: log, location files and connection to MIT geocoder API
output: {wifi id, bldg, room, lon, lat}

function PrepareData
     for line in wifi log file
         split timestamp, count, wifi ap id
     return list of items

load json file "location.json"
     join wifi ap id to bldg name and room
     connect to MIT map geocoder API
     for location, id in locations
           get bldg, room strings
           get lat, lon as x,y
     print (wifi ap id,bldg,room,lon,lat)
```

The final dataset once scraped and merged contains the fields: time, room, number of connected devices, along with latitude and longitude which plots to the following map in Figure 6.12:

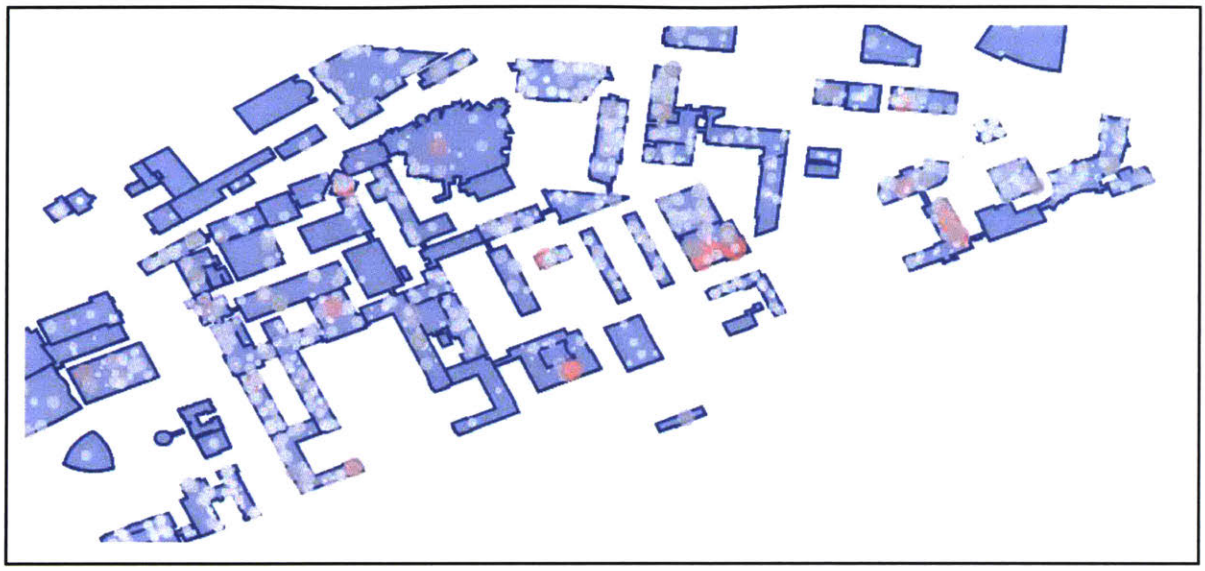


Figure 6.12 A polygon shape map of MIT

This chapter walks through the preparation techniques for data mining and cleansing large image and WiFi datasets with the assistance of machine learning algorithms to accelerate the image classification process. In the next chapter, the clean datasets become the necessary component for visualizing large amounts of data with custom built software tools.

Chapter 7

Summarizing Data for Visualization

It is useful to plot and visually understand the underlying structure of a dataset that is in the form of numbers. Here, a graphical summary can yield insights about the data distribution, variable relationships, and the quantification of the spread.[28] Other times, however, a standard method for displaying information does not exist, as is the case when plotting 3D camera poses extracted from 2D imagery. Often times, the challenge in this research is taking different types of very large datasets, combining them together, and finding the best way to display content that is visually comprehensive for actionable analysis. This chapter discusses the decision making process on creating a visualization tool, only to surprisingly discover that in my discussion with potential users, only some software features pique their interests more than others.

In the AEC world, various types of communication platforms exist in addition to the uses of CAD and Building Information Modeling (BIM) systems. Traditional reporting tools like spreadsheets, still serve as the business intelligence tool in the AEC industry[29] (Figure 7.1). The disadvantages of using spreadsheets when a lot of information exists can impede actionable

[28] Rice, J.A. Mathematical Statistics and Data Analysis, 3rd ed. Thomson, Belmont: 2007.
[29] Vanderzyle, Bruce. "Business Intelligence in the Construction Industry". Presented at the Chapter of the Construction Financial Management Association (CFMA) of Massachusetts.

decisions when the data can be hard to read and cite what is important. On top of that, if a user needs more reports, diving in for information can be an arduous process. The traditional BIM model that AEC professionals use typically contain information that is project related without access to information outside the walls of the profession.

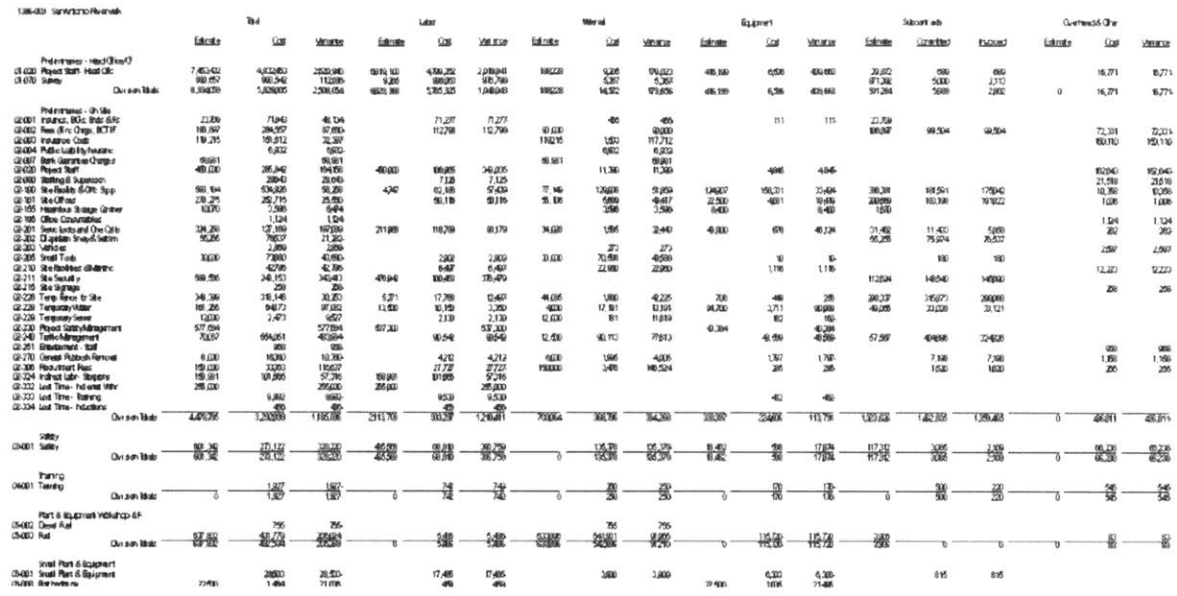


Figure 7.1 Spreadsheets as reporting technology in AEC.[30]

7.1 Extraction of Multiple Camera Poses from the Web

Exploring thousands of data can prove challenging without a strategic way to view the information. Traditional bar charts and scatter plots can provide a slice of information and understanding, yet large datasets can be complex. The goals for creating the user interface in this research is to let users explore a combination of spatial and non-spatial attributes such as location, color, and so on. The means in which a user can view the information also matters, so building the system for the web becomes the platform of choice because of its relative ease of accessibility to the user base. The other goal involves designing an interface that can present information in a clutter-free way with the following information to show users: the original camera position taken from 2D images relative to its location on a floor plan, filtering each camera object

[30]http://cafe.cfma.org/MassBostonMA/resources/resources/viewdocument/?DocumentKey=885d079c-eceb-41af-9983-b7952182b87b

by the year taken illustrating some key cluster locations by the public since a building's opening. Showing cameras filtered by year, lends some insight as to which part of a building facade does the general public find most memorable in addition to knowing where pedestrian traffic builds up to photograph a building. Other types of visualization include the presentation of 3D point cloud information, to be discussed in Section 7.3.

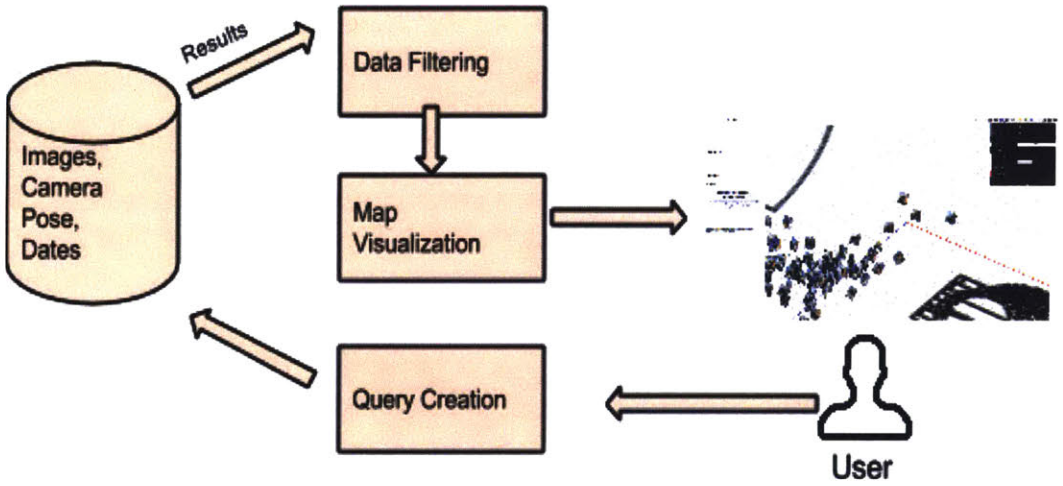


Figure 7.2 User interaction workflow

Engaging with three-dimensional information is common in AEC from 3D virtual models and sketches. The creation of a 3D visualization and analysis tool for this research comes as a natural extension. Figure 7.2 shows how a user can query the system by date and the results undergo a filtering process to output original images and camera positions above a building floor plan. Figure 7.3 illustrates the web interface with three pane windows for querying, viewing, and control manipulation. Panel A allows a user to toggle multiple cameras on/off and filter cameras and images by year, Panel B shows the main view containing the 3D point cloud, camera poses and their accompanying images above a floor plan, while Panel C allows a user to adjust the 3D environment view, grid, and colors along other control-like features. The web application begins by presenting a global view of MIT Stata Center - allowing users to cite camera clusters easily. In Figure 7.4, a group of cameras from 2006 until 2010 lay above Stata Center's outdoor amphitheater (circled in red). Upon closer inspection, people took pictures of Stata Center at

nearly the same location (Figure 7.5 and 7.6) and were drawn to a particular part of the exterior facade (Figure 7.7).

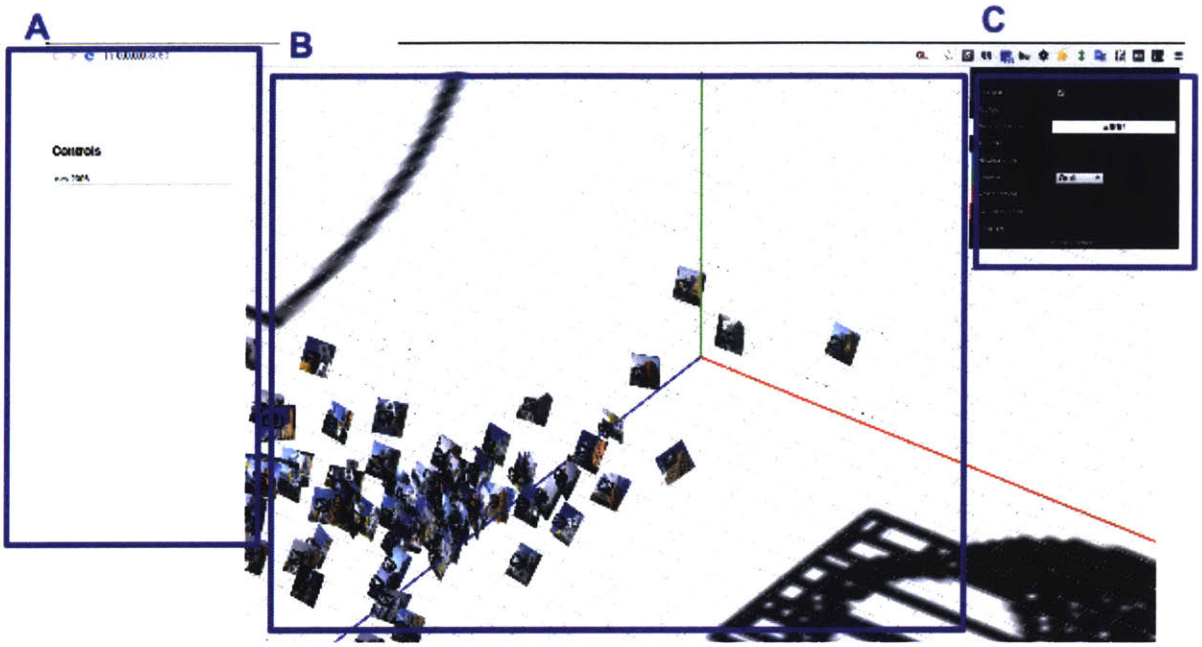


Figure 7.3 The web interface to view 3d point clouds and camera position filtered by date.

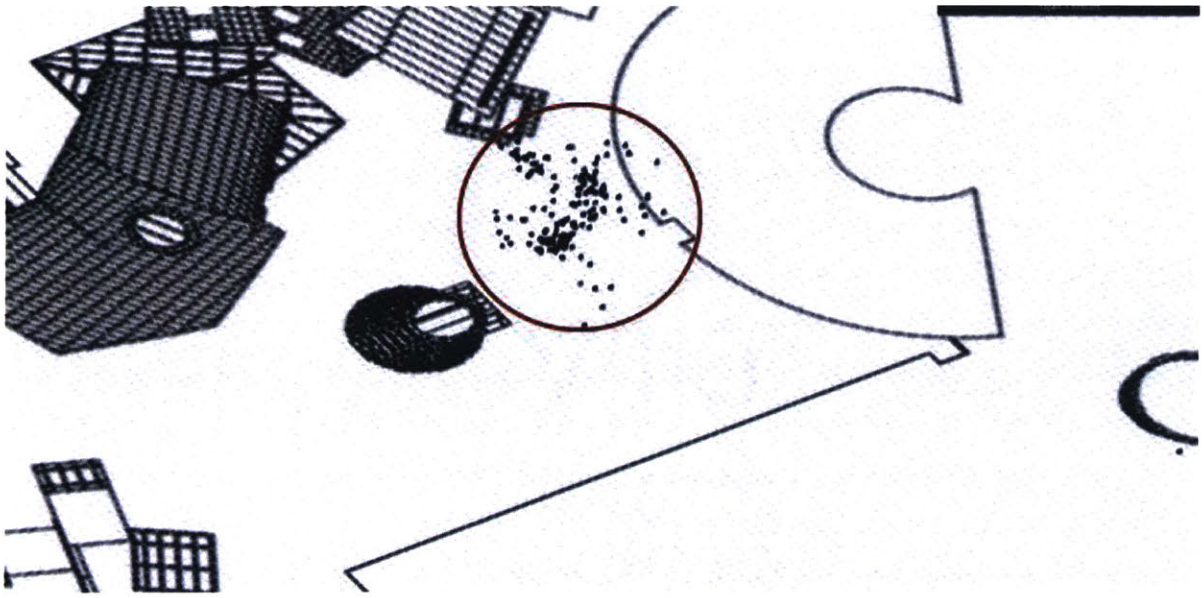


Figure 7.4 A global level of detail.

Figure 7.5 A cluster of camera objects in wireframe

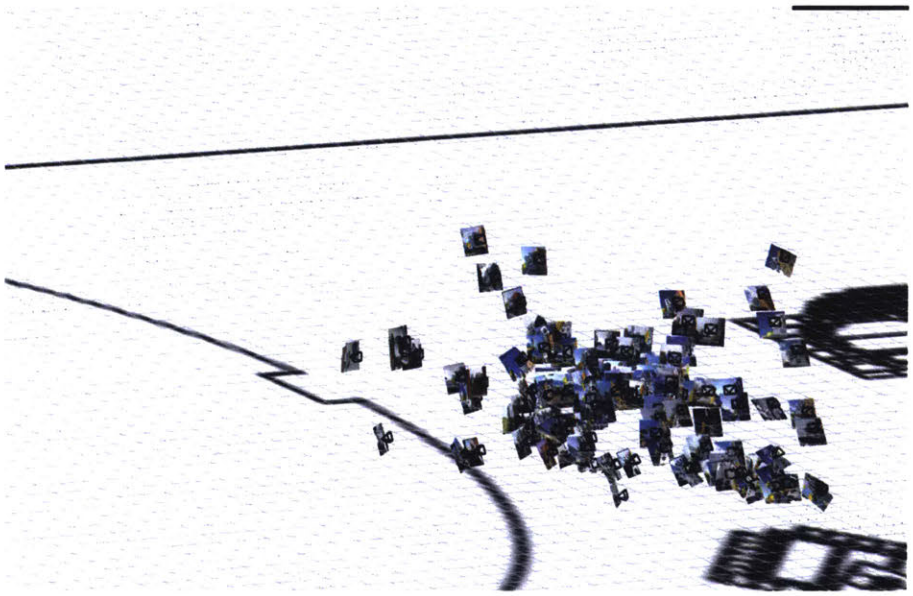


Figure 7.6 A cluster of camera objects with the original image shots taken.

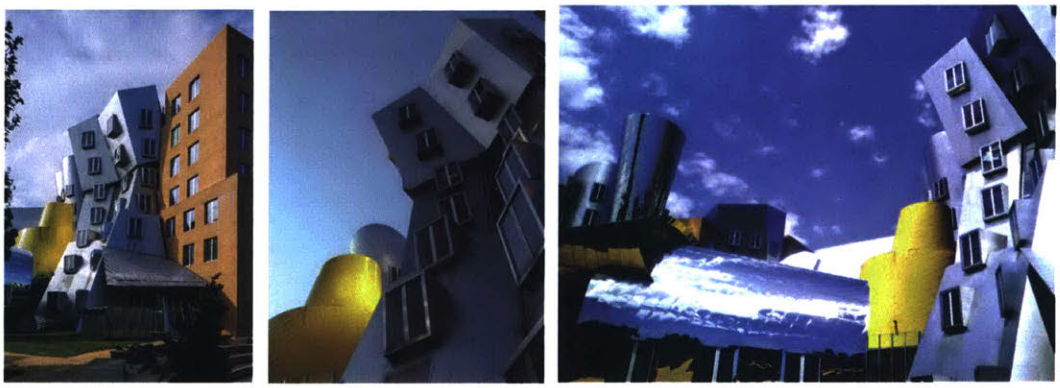


Figure 7.7 Sample of images taken from the camera cluster in figure 7.5 and 7.6.

7.2 Image-based Point Cloud

The majority of this chapter talks about the extraction of camera positions from 2D images, yet the images can also be used for generating a point cloud model, or a set of data points in a three-dimensional coordinate system[31]. A point cloud is usually converted into a polygon mesh model for surface reconstruction, but the application developed for this dissertation leaves the visualization at the point cloud stage -- the density of some of the point clouds for Stata Center, for example, exceeds over 8 million points which is enough to view a comprehensive visual inspection of the building represented as points. Having eight million points, however, is considered "large data" for a web application, and must undergo down sampling to roughly around two million points. Figure 7.8 shows the photographs that were used to process the point cloud in Figure 7.9.

The usefulness in creating a 3D point cloud model serves two purposes: 1) To understand how a building's facade has changed since post-construction and 2) When placing the camera pose object relative to the 3D point cloud, the user (of the web platform) can understand what particular building scene capture the public's eye. Again, knowing what particular scene or part of a building resonates with the public is useful for a city administrator, so as to allocate spending and other resources on the city icons that attract visitors thereby generating tourism revenue. For planners and designers, knowing what buildings attract crowds can help aide in the future design of public transit stations and facilities. For a historian, knowing where individuals clusters to photograph a city icon can demonstrate how the public sees what buildings are representative of a city throughout time. Other buildings analyzed for comparison in this study include Simmons Hall (Figure 7.10) and Copley Square (Figure 7.11).

[31] https://en.wikipedia.org/wiki/Point_cloud

Figure 7.8 (above) Original images used for point cloud generation and the extraction of original camera positions.

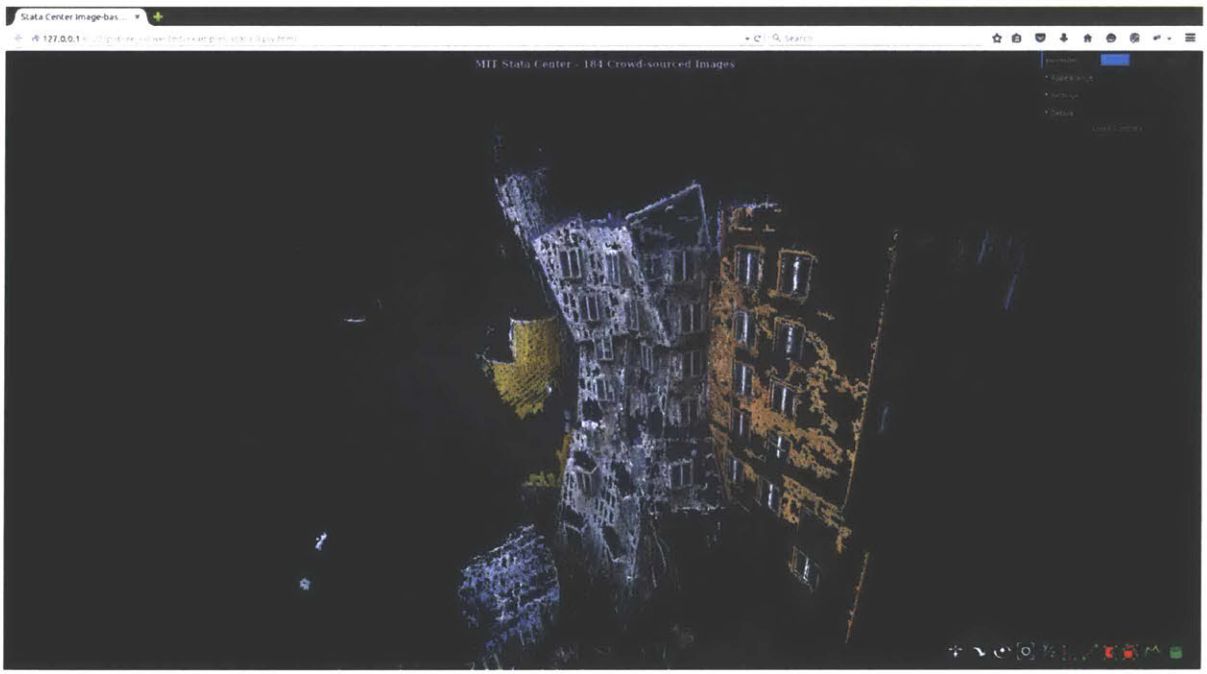


Figure 7.9 (a) Stata Center point cloud facade from 184 crowd-sourced images

Figure 7.9 (b) Stata Center on the side of Vassar Street as seen from 300 crowd-sourced images

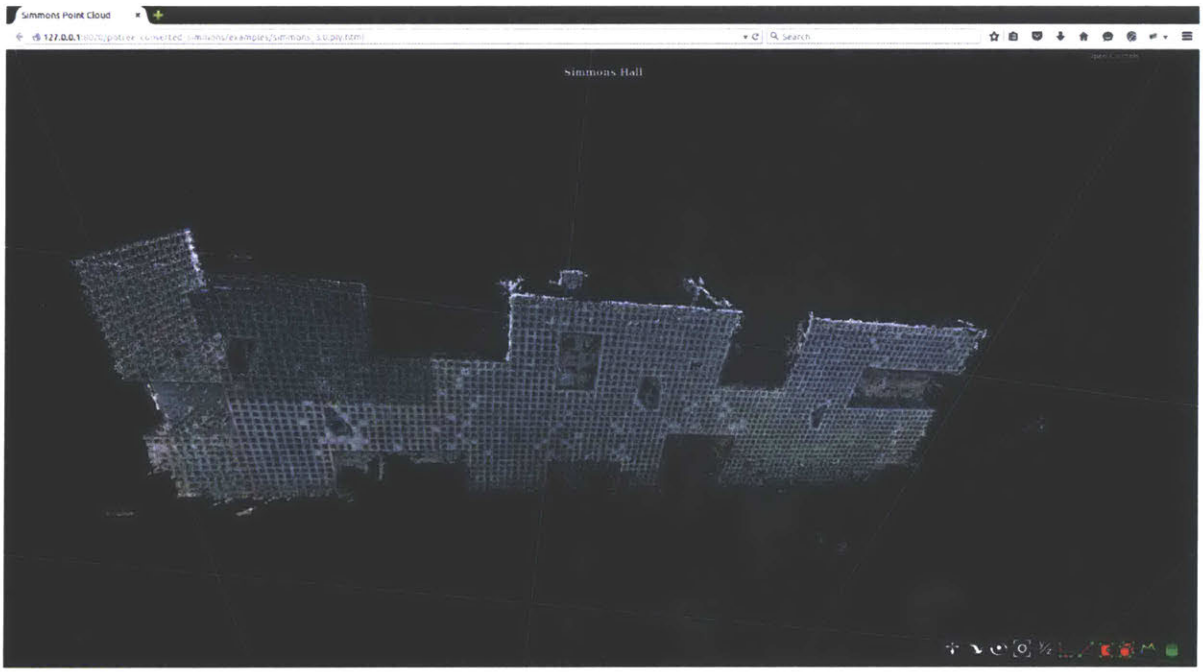


Figure 7.10 Simmons Hall at MIT from analyzing 1,324 crowd-sourced images

Figure 7.11 129, 855,598 points were processed and 119,570 points (13.97%) were written as output in 5.42 seconds.

Chapter 8

Conclusion and Future Work

This thesis gathers streamable data from the web to explore and propose the integration of non-traditional data sets for the AEC/FM/Historian management, design, or documentation process. The importance of this work highlights a new understanding of built works in post construction to inform future discipline relevant decisions and explore alternative ways to understand information related to the built environment. In order to materialize the non-traditional data analysis, a software pipeline was developed that aims to ease the process of data mining, extraction, and analysis of crowd sourced data: public photographs, text, and (private) WiFi Access Point connectivity.

Such an endeavor resulted in a discussion with facilities management staff and a historian, they confirmed the potential of the research deliverables for the following applications: campus security/crowd control and understanding how publicly composed images of buildings influence architectural design, respectively. Other areas where this research finds applicability suggests the possibility for future work -- an integration between crowd-sourced public data and Building Information Models (BIM). With the generation of an image based 3D model from public photographs, the image based model can become substantial enough to become the BIM model that contains information beyond the building's initial construction.

Such a BIM system can also integrate data analysis and multiple streams of crowd-sourced information inside and outside of an organization as an integral part of the design decision

analysis, management of a building, or for learning about a building's historical record as seen through the eyes of the public. Currently, creating a BIM model does not require or contain any information gathered from publicly available sources outside of itself with the exception of BIM model components (walls, flooring, doors, etc) from public repositories online. Data sources from image databases, local governments, current climate, finance, consumer, manufacturing, and other resources that could facilitate a dialogue in the design and construction process from within the BIM environment has yet to be explored.

There is a massive volume of content, if not clutter. Despite the faddish elements of "big data," the opportunity comes from analyzing and combining the data to extract insights and value[32]. With all of this information at our disposal and as the increasing penetration of Building Information Modeling (BIM) into the AECFM industry continues, the tools produced in this research challenges the current controlled and authorized BIM environment and instead, suggests a BIM environment that contains a fusion with up-to-date information generated by the people who occupy the building.

The future work of this research leans toward the exploration of a new kind of building information model, one that integrates data analysis and multiple streams of crowd-sourced information inside and outside of an organization as an integral part of the design decision analysis, management of a building, or for learning about a building's historical record as seen through the eyes of the public. Consequently, such an integration adds to the "information" in building information modeling by including public APIs not traditionally brought into the design and management process in AEC and related fields. The works described herein already explore the use of public and private data sets for the use case personas who already engage with BIM or 3D models to some capacity.

This research also explores algorithms for machine learning and computer vision for automating the extraction of unsatisfactory images in the data collection -- executing from example data or past experience. A model defined some parameters using training data/past experience to gain knowledge and make predictions in the future. Further exploration and integration of machine learning and computer vision for this research includes training an algorithm to classify the data

[32] In this research, "value" is relative to the goals of each use case persona.

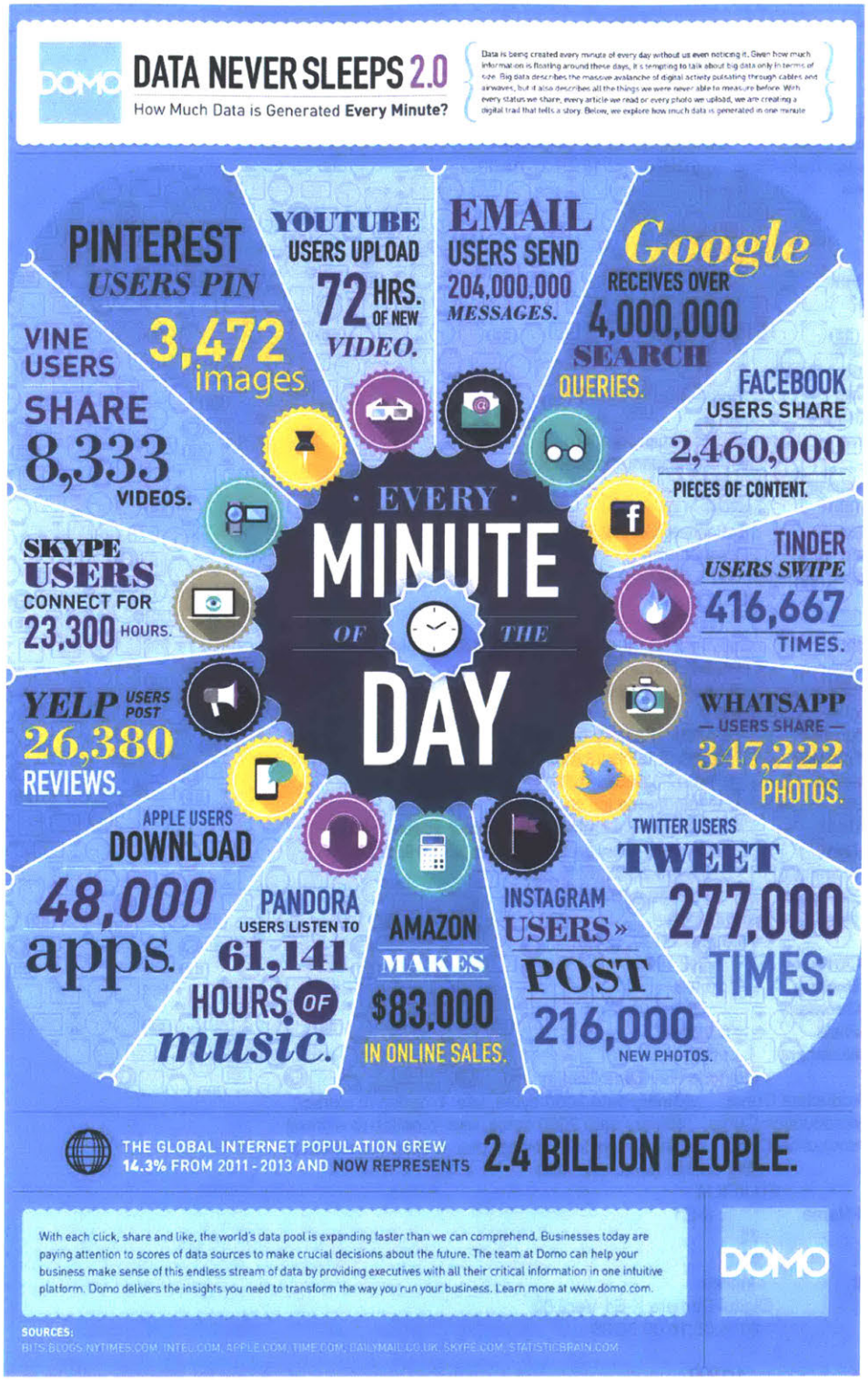
analysis. For instance, the algorithm learns the spatial point patterns of camera positions and WiFi access point connectivity, then proposes a semantic understanding of the data patterns.

Through the use case scenario of a designer/city administrator, the analysis produced in this research allows for the discovery of building types and facades that attract visitors as well as track camera positions/foot traffic from internet photographs over time. In turn, this can facilitate the planning of future building projects for economic growth -- bus stops, taxi stands, retail, and parking; for facilities management, the WiFi access point data was analyzed to determine where crowds form and when to anticipate crowds at MIT with predictive analytics as well as using the extracted camera positions.

Future work in this area include revisiting the WiFi data to track individual anonymous users access point connectivity pattern so as to further understand how individuals occupy buildings throughout the day. The 3D point cloud models generated from the crowd sourced image dataset can also allow a facility manager to learn how a building has changed since post production due to time and weather by analysing photographs for comparison; for historians, this research looks at an array of images and extracts their original camera positions, giving historians the ability to understand how the public perceives buildings through photographs and their selective geometric point of view. The extraction of textual data in addition to photographs also lends to understanding the public's interpretation of built works over time.

As the public generates more information saved into public repositories, that information's programmatic accessibility warrants an exploratory analysis that can best suit the purposes for decision making in applications for architecture, engineering, and construction and related communities. The current frameworks and methodologies in the selected disciplines, as this research suggests, do not integrate an amalgam of crowdsourced public data into their workflows. All in all, this thesis describes the possibilities for including non-traditional big data to support, document, and infer future results for assisting the design, documentation, and management process for AEC and related industries.

Appendix A: Generating Data by the Minute



Appendix B: EXIF Data

```
File Name                       : stata99.jpg
Directory                       : .
File Size                       : 3.6 MB
File Modification Date/Time     : 2015:05:02 13:15:21-04:00
File Access Date/Time           : 2015:06:05 16:50:04-04:00
File Inode Change Date/Time     : 2015:05:02 13:15:21-04:00
File Permissions                : rw-------
File Type                       : JPEG
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
Profile CMM Type                : Lino
Profile Version                 : 2.1.0
Profile Class                   : Display Device Profile
Color Space Data                : RGB
Profile Connection Space        : XYZ
Profile Date Time               : 1998:02:09 06:49:00
Profile File Signature          : acsp
Primary Platform                : Microsoft Corporation
CMM Flags                       : Not Embedded, Independent
Device Manufacturer             : IEC
Device Model                    : sRGB
Device Attributes               : Reflective, Glossy, Positive, Color
Rendering Intent                : Perceptual
Connection Space Illuminant     : 0.9642 1 0.82491
Profile Creator                 : HP
Profile ID                      : 0
Profile Copyright               : Copyright (c) 1998 Hewlett-Packard Company
Profile Description             : sRGB IEC61966-2.1
Media White Point               : 0.95045 1 1.08905
Media Black Point               : 0 0 0
Red Matrix Column               : 0.43607 0.22249 0.01392
Green Matrix Column             : 0.38515 0.71687 0.09708
Blue Matrix Column              : 0.14307 0.06061 0.7141
Device Mfg Desc                 : IEC http://www.iec.ch
Device Model Desc               : IEC 61966-2.1 Default RGB colour space - sRGB
Viewing Cond Desc               : Reference Viewing Condition in IEC61966-2.1
Viewing Cond Illuminant         : 19.6445 20.3718 16.8089
Viewing Cond Surround           : 3.92889 4.07439 3.36179
Viewing Cond Illuminant Type    : D50
Luminance                       : 76.03647 80 87.12462
Measurement Observer            : CIE 1931
Measurement Backing             : 0 0 0
Measurement Geometry            : Unknown (0)
Measurement Flare               : 0.999%
Measurement Illuminant          : D65
Technology                      : Cathode Ray Tube Display
Red Tone Reproduction Curve     : (Binary data 2060 bytes, use -b option to extract)
Green Tone Reproduction Curve   : (Binary data 2060 bytes, use -b option to extract)
Blue Tone Reproduction Curve    : (Binary data 2060 bytes, use -b option to extract)
Exif Byte Order                 : Big-endian (Motorola, MM)
Make                            : FUJIFILM
Camera Model Name               : X-E1
X Resolution                    : 72
Y Resolution                    : 72
Resolution Unit                 : inches
Software                        : Digital Camera X-E1 Ver2.00
Modify Date                     : 2014:05:18 09:30:36
Copyright                       :
Exposure Time                   : 1/1900
F Number                        : 6.4
```

Exposure Program : Aperture-priority AE
ISO : 400
Sensitivity Type : Standard Output Sensitivity
Exif Version : 0230
Date/Time Original : 2014:05:18 09:30:36
Create Date : 2014:05:18 09:30:36
Components Configuration : Y, Cb, Cr, -
Compressed Bits Per Pixel : 3.2
Shutter Speed Value : 1/2006
Aperture Value : 6.5
Brightness Value : 10.19
Exposure Compensation : +0.33
Max Aperture Value : 2.8
Metering Mode : Multi-segment
Light Source : Unknown
Flash : Off, Did not fire
Focal Length : 18.0 mm
Flashpix Version : 0100
Color Space : sRGB
Exif Image Width : 3264
Exif Image Height : 4896
Focal Plane X Resolution : 2092
Focal Plane Y Resolution : 2092
Focal Plane Resolution Unit : cm
Sensing Method : One-chip color area
File Source : Digital Camera
Scene Type : Directly photographed
Custom Rendered : Normal
Exposure Mode : Auto
White Balance : Auto
Focal Length In 35mm Format : 27 mm
Scene Capture Type : Standard
Sharpness : Normal
Subject Distance Range : Unknown
Lens Info : 18-55mm f/2.8-4
Lens Make : FUJIFILM
Lens Model : XF18-55mmF2.8-4 R LM OIS
Lens Serial Number : 24A50268
XMP Toolkit : XMP Core 5.4.0
Lens : XF18-55mmF2.8-4 R LM OIS
Creator Tool : Digital Camera X-E1 Ver2.00
Date Created : 2014:05:18 09:30:36
Rights :
Image Width : 3264
Image Height : 4896
Encoding Process : Baseline DCT, Huffman coding
Bits Per Sample : 8
Color Components : 3
Y Cb Cr Sub Sampling : YCbCr4:2:0 (2 2)
Aperture : 6.4
Image Size : 3264x4896
Scale Factor To 35 mm Equivalent: 1.5
Shutter Speed : 1/1900
Circle Of Confusion : 0.020 mm
Field Of View : 67.4 deg
Focal Length : 18.0 mm (35 mm equivalent: 27.0 mm)
Hyperfocal Distance : 2.53 m
Light Value : 14.2

Appendix C: Statistical Reasoning

This section describes the statistical methods used for analyzing the following datasets: camera pose extraction from 2D images, EXIF metadata from original images, WiFi access timestamps and device counts, and photo captions. While a vast array of statistical models exist, only a handful become useful depending on the type of data and its underlying model fit - linear or non-linear, time series, gaussian/non-gaussian data, and others. The primary goals for employing the statistical methods herein is to learn from the data and to develop an estimation for predicting near future values.

Time Series Components

Before applying time series analysis methods, a linear regression is generally a simple general-purpose tool to get started. There are four components to a time series:

Trend

A smooth function that captures the direction of a time series.

Cyclical Variation

Rise and fall over a periods longer than one year.

Seasonal Variation

A time series pattern of change within a year that is often repeatable.

Irregular Variation

An unpredictable fluctuation in time series data (i.e. the initial economic impact of war).

Regression is one way to extract a trend from time series information, but if the trend is not linear, an alternative is to use a **moving average** and/or a **weighted moving average**. The moving average smooths the fluctuation in the data rather than using the least squares method, $Y = a + bt$. Applying a moving average to a time series,

Estimation of the Correlation Coefficient

A correlation coefficient measures the direction and strength of a linear relationship between two or more variables (and nonlinear relationships with maximal correlation). The most common linear correlation coefficient is the Pearson Product Moment Correlation Coefficient, the mathematical formula for computing r is:

$$r = \frac{n\Sigma xy - (\Sigma x)(\Sigma y)}{\sqrt{n(\Sigma x^2) - (\Sigma x)^2}\sqrt{n(\Sigma y^2) - (\Sigma y)^2}}$$

where n is the number of pairs of data.

Equation 5.1

The value of r is such that $-1 \leq r \leq +1$. In other words, a value close to +1 indicates a strong positive linear correlation where the x and y variables both increase. An r value of -1 indicates a negative linear correlation such that the x values increase as the y values decrease. If r = 0 or near 0, a weak linear correlation/no correlation exists between variables indicating a random relationship. Figure 5.17 demonstrates the correlation coefficient as plotted between x and y variables.

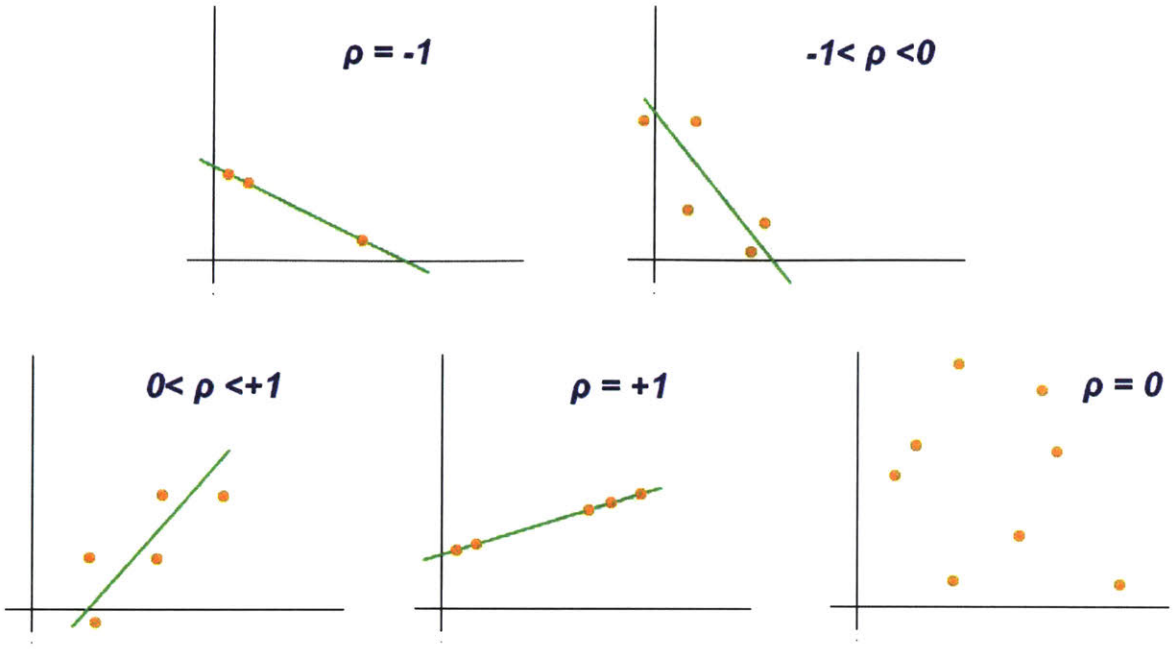


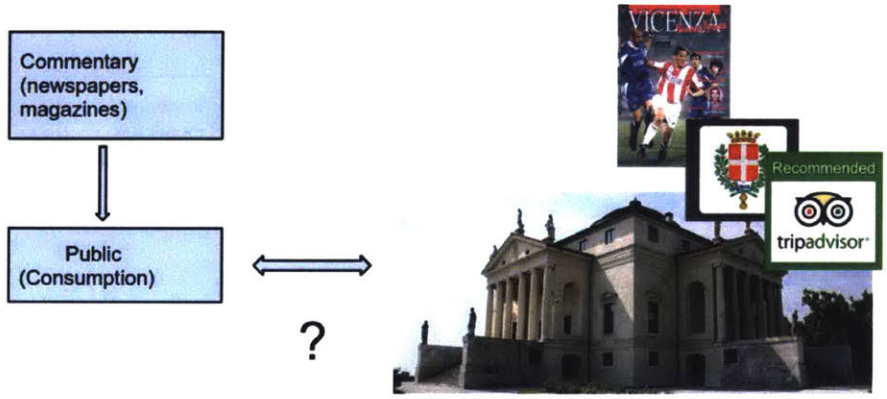
Figure 5.17. Pearson correlation coefficient.[33]

Equally significant is the Coefficient of Determination, r^2 or R^2, or square of the correlation, because r^2 measures the fluctuation of variables between one another and indicates how well data fits to a statistical model, or the goodness of fit. For instance, if $r = 0.922$ and $r^2 = 0.850$, then 85% of the total variation (fluctuation) in the dependent variable (y) can be explained by the linear relationship between x and y. The remaining 15% of the total variation is unknown. Similar to the correlation coefficient, r^2 ranges from 0 to 1. An r^2 of 0 indicates that the dependent variable (y) cannot be "predicted" from an independent variable (x) because the (regression) line does not fit the data. Likewise, an r^2 of 1 means that y can be predicted without error from x because the line fits the data well.

[33] "Correlation coefficient" by Kiatdd - Own work. Licensed under CC BY-SA 3.0 via Commons - https://commons.wikimedia.org/wiki/File:Correlation_coefficient.png#/media/File:Correlation_coefficient.png

Appendix C: Text Mining and Natural Language Processing with Twitter Case Study

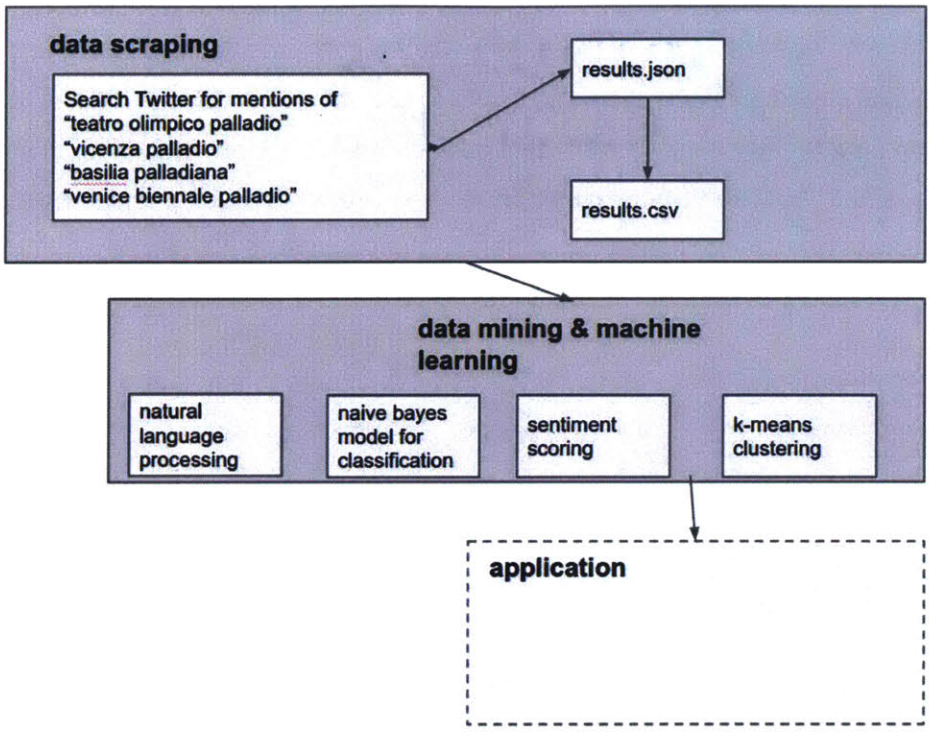
This case study was produced as a final project in the MIT Palladio Workshop course taught by Professor Takehiko Nagakura, Visiting Researcher Daniel Tsai, and Professor Howard Burns of Spring 2014. This project explores the process of value creation through the discourse from content creators, intermediaries (commentators such as the NY Times Arts), and consumers on Twitter. The goal is 1) to observe what commentators are saying about a particular popular



attraction from a positive or negative construction of meaning; 2) to observe whether consumers are reacting negatively or positively to the intermediaries; 3) observe how sites portray themselves to consumers on Twitter followed by consumer reaction. This paper examines a portion of the value chain in the creative/cultural industries in terms of the creators, trade intermediaries, and consumers.

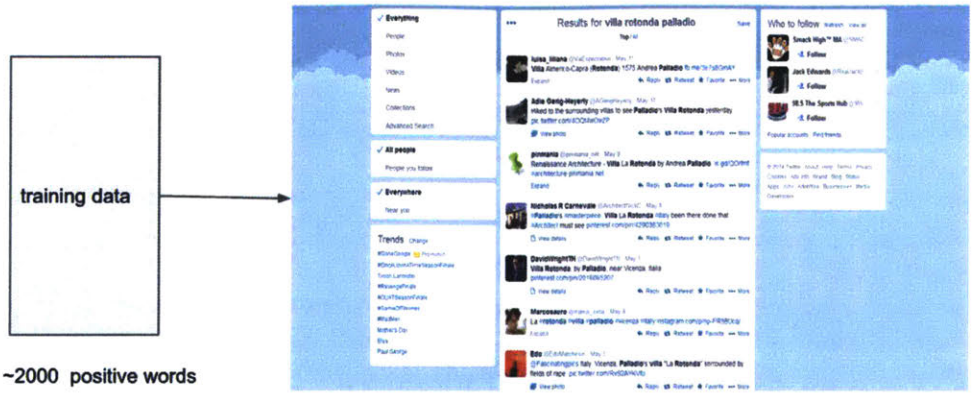
The process of extracting information from Twitter began by scraping Twitter feeds programmatically through Twitter's Application Programming Interface (API), the rate limit of calls to Twitter's server is 150 calls per hour or 150 tweets/hour which limits the amount of information one can extract. The data scraping process begins by searching for mentions of "vicenza," "from user: @vicenza" and "palladio from:basillica." The first search returned 100 tweets mentioning gerhard richter; the second returned tweets from the artist's twitter profile, and lastly, the third search returns tweets about the artist mentioned by the New York Times

Arts. Two methods were explored for analysis and validation, sentiment scoring and Naive Bayes while R and Python were used interchangeably.



Sentiment Scoring

To determine the sentiment of each tweet, or whether the tweet contains positive or negative meaning. The process begins by storing each json-format search results and converting to csv format, then inputting the data into the extraction pipeline, Figure 1.



The key task of interpreting natural language text is to extract words and determine its sentiment orientation aided by the sentiment scoring algorithm to classify whether a word is positive or negative. In this case, a freely distributed opinion lexicon[34] by Hu and Liu consists of 2,006 positive words (e.g. accomplish, acclaim, abundant) and 7,783 negative words (e.g. irrelevant, abnormal, disservice) were used for estimating sentiments. For example, the sentence:

"Experiencing the 16thC architectural delights of Andrea Palladio & his sublime, good-looking villa rotonda."

score = 3

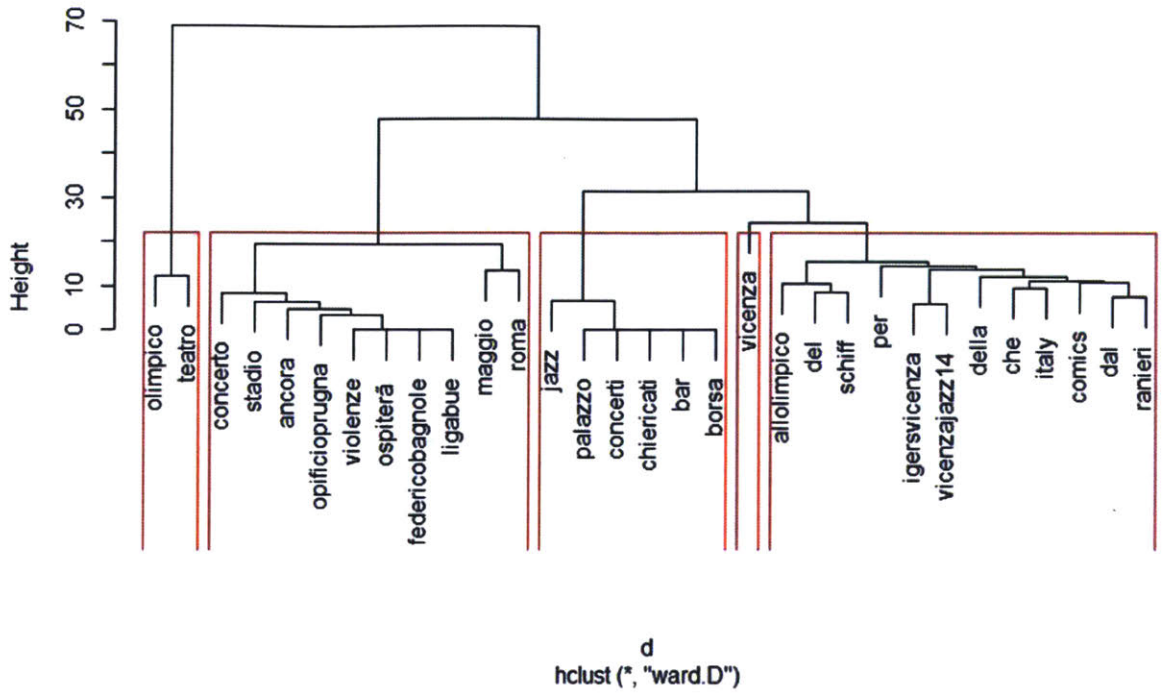
The words in green colored font are classified as positive and each contribute to one point, totalling towards 3 points. Another example:

[34] Ding, Xiaowen, Bing Liu, and Philip S Yu. "A holistic lexicon-based approach to opinion mining." *Proceedings of the 2008 International Conference on Web Search and Data Mining* 11 Feb. 2008: 231-240.

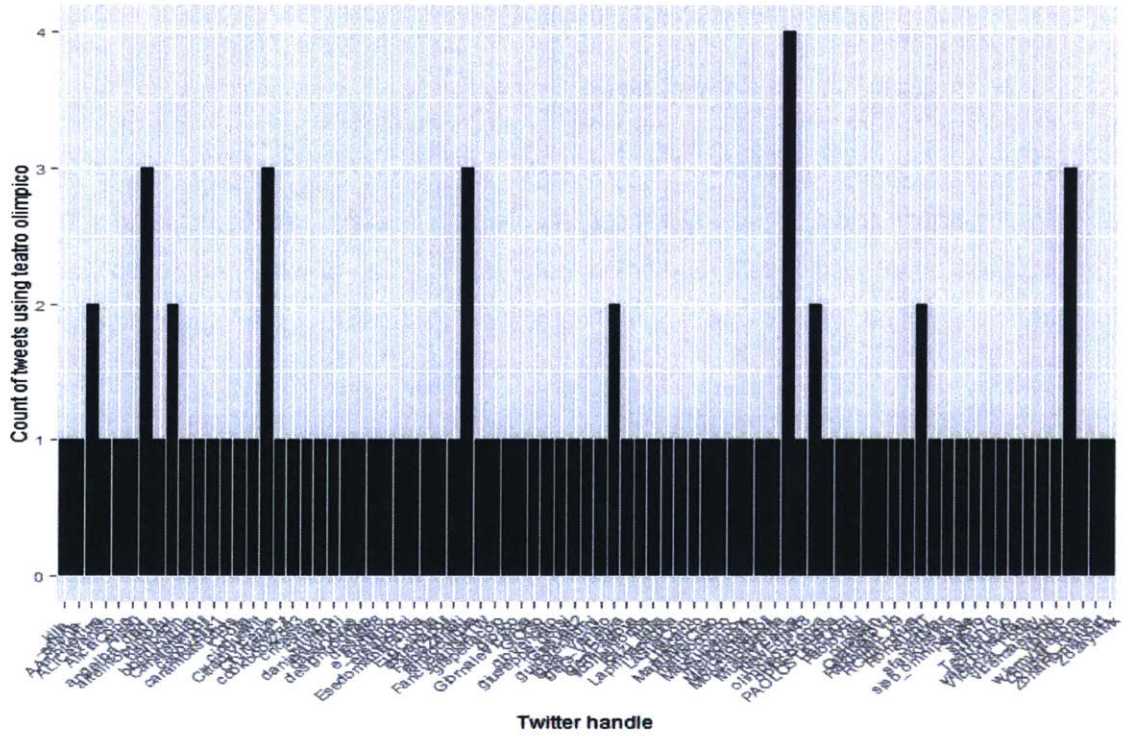
"Crazy lines at Teatro Olimpico, delayed to go see it. #annoying"

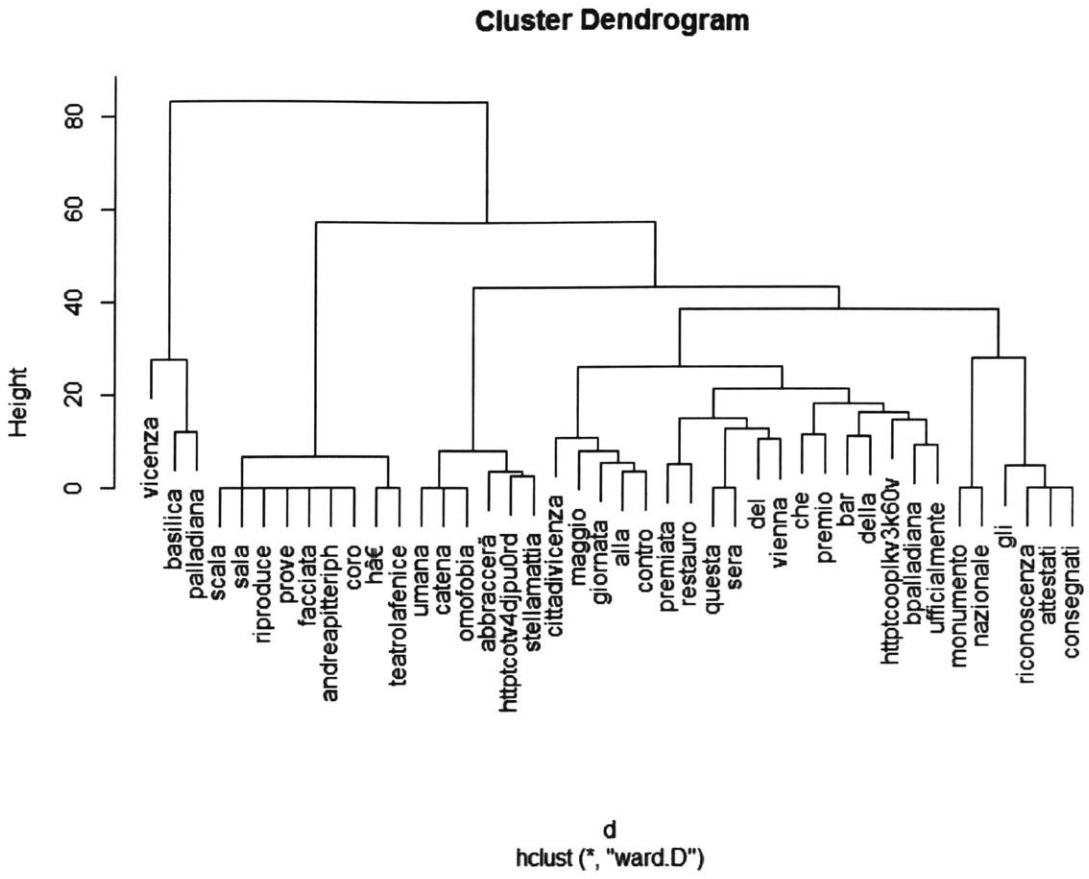
score = -3 1

The font in red contribute to three negative scores, -3, with one positive green score, +1, totalling to -2. The total score of -2 is classified as a negative comment. Further exploration into how each word relates to one another was measured by Euclidean distance using the k-means algorithm in order to describe a cluster of words that generally belong to one another. This is useful for exploring words that are associated to one another from tweets.



cluster analysis of "teatro olimpico" measured in euclidean distance



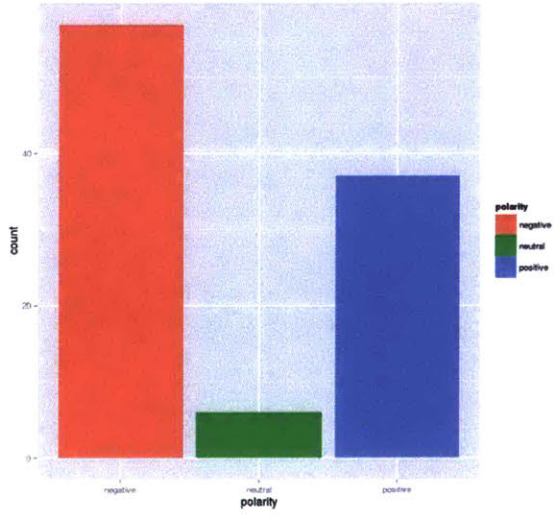


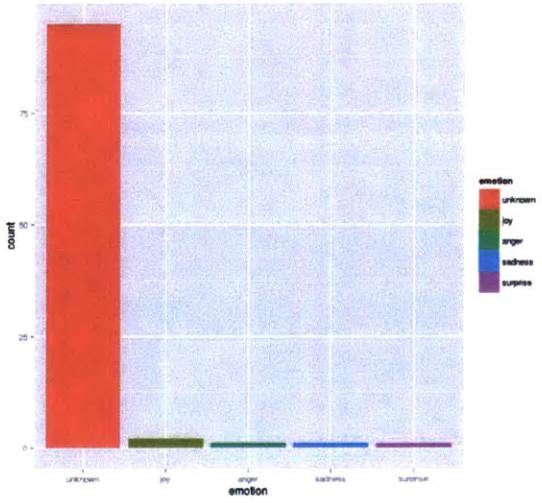
Naive Bayes Method for Classification

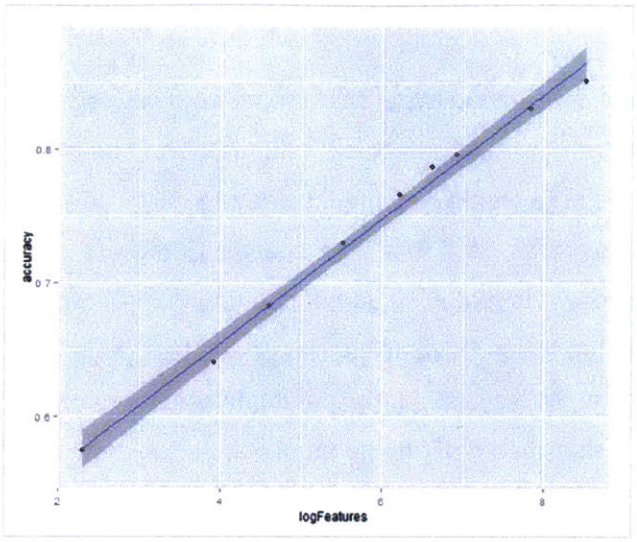
The Naive Bayes classifier determined tweet sentiments from labeled features (training set) and made assumptions on unseen tweets (unlabeled features). The bag-of-words approach was used for feature extraction, or to create a numerical feature vector after cleaning up the tweets from numbers, punctuation marks, html links, and spaces. Some of the limitations of this approach is that the bag-of-words model could not capture multiple word expressions or phrases. The accuracy of the Naive Bayes classifier on the test set is 74%. Contrary to the sentiment scoring algorithm, Bayes ranked most of the tweets as negative in the polarity diagram of Figure 5a and classified most of the emotions as "unknown" in Figure 5b. Increasing the accuracy of the classifier can be done by inserting a larger training set, or increasing the features selection as cited in an example of movie reviews where using 10,000 and 15,000 features increases the accuracy to ~85%.

Naive Bayes Method for Classification

The Naive Bayes classifier determined tweet sentiments from labeled features (training set) and made assumptions on unseen tweets (unlabeled features). The bag-of-words approach was used for feature extraction, or to create a numerical feature vector after cleaning up the tweets from numbers, punctuation marks, html links, and spaces. Some of the limitations of this approach is that the bag-of-words model could not capture multiple word expressions or phrases. The accuracy of the Naive Bayes classifier on the test set is 74%. Contrary to the sentiment scoring algorithm, Bayes ranked most of the tweets as negative in the polarity diagram of Figure 5a and classified most of the emotions as "unknown" in Figure 5b. Increasing the accuracy of the classifier can be done by inserting a larger training set, or increasing the features selection as cited in an example of movie reviews where using 10,000 and 15,000 features increases the accuracy to ~85%







A linear relationship between the accuracy and log of number of features.
Source: "Andy Bromberg." *Sentiment Analysis*. Web. 13 May 2014.

Appendix D: Camera Types and Frequency

Copley Square			Stata Center		
	x	freq		x	freq
1	1	1	1	-	163
2	Apple	27	2	Apple	89
3	Canon	208	3	Canon	491
4	CASIO	2	4	CASIO COMPUTER CO.,LTD	1
5	CASIO COMPUTER CO.,LTD.	8	5	CASIO COMPUTER CO.,LTD.	8
6	Cosina	1	6	EASTMAN KODAK COMPANY	47
7	EASTMAN KODAK COMPANY	5	7	FUJIFILM	55
8	FUJI PHOTO FILM CO., LTD.	4	8	Hewlett-Packard	3
9	FUJIFILM	3	9	Hipstamatic	1
10	KONICA MINOLTA	1	10	HTC	1
11	LEICA	8	11	LEICA	7
12	motorola	1	12	made by Polaroid	8
13	NIKON	19	13	Minolta Co., Ltd.	8
14	NIKON CORPORATION	88	14	Motorola	3
15	NORITSU KOKI	1	15	NIKON	50
16	OLYMPUS CORPORATION	1	16	NIKON CORPORATION	257
17	OLYMPUS IMAGING CORP.	18	17	Nokia	21
18	OLYMPUS OPTICAL CO.,LTD	2	18	OLYMPUS CORPORATION	2
19	Panasonic	6	19	OLYMPUS IMAGING CORP.	10
20	PENTAX	1	20	OLYMPUS OPTICAL CO.,LTD	2
21	SAMSUNG	1	21	Panasonic	65
22	SAMSUNG TECHWIN	1	22	PENTAX	1
23	SONY	20	23	PENTAX Corporation	5
24	Vignette	5	24	Research In Motion	1
			25	SAMSUNG	3
			26	SONY	88
			27	Sony Ericsson	6
			28	Vignette	2

Simmons Hall

	x	freq
1	-\t-	128
2	-\tCanon EOS 40D	1
3	-\tCanon EOS DIGITAL REBEL XSi	2
4	1\tSGH-T139	1
5	Apple\tiPhone	1
6	Apple\tiPhone 3G	5
7	Apple\tiPhone 3GS	21
8	Canon\tCanon DIGITAL IXUS 900Ti	3
9	Canon\tCanon DIGITAL IXUS v	2
10	Canon\tCanon EOS 30D	3
11	Canon\tCanon EOS 350D DIGITAL	2
12	Canon\tCanon EOS 40D	2
13	Canon\tCanon EOS 500D	12
14	Canon\tCanon EOS 5D	6
15	Canon\tCanon EOS 5D Mark II	40
16	Canon\tCanon EOS 5D Mark III	1
17	Canon\tCanon EOS DIGITAL REBEL XT	1
18	Canon\tCanon EOS DIGITAL REBEL XTi	16
19	Canon\tCanon EOS REBEL T1i	18
20	Canon\tCanon EOS REBEL T2i	2
21	Canon\tCanon EOS REBEL T3i	1
22	Canon\tCanon PowerShot A530	3
23	Canon\tCanon PowerShot A550	1
24	Canon\tCanon PowerShot A560	2
25	Canon\tCanon PowerShot A570 IS	1
26	Canon\tCanon PowerShot A610	5
27	Canon\tCanon PowerShot A630	4
28	Canon\tCanon PowerShot A710 IS	5
29	Canon\tCanon PowerShot A80	1
30	Canon\tCanon PowerShot G10	2
31	Canon\tCanon PowerShot G9	4
32	Canon\tCanon PowerShot S20	5
33	Canon\tCanon PowerShot S400	13
34	Canon\tCanon PowerShot S5 IS	6
35	Canon\tCanon PowerShot SD300	2
36	Canon\tCanon PowerShot SD450	4
37	Canon\tCanon PowerShot SD750	2
38	Canon\tCanon PowerShot SD800 IS	1
39	Canon\tCanon PowerShot SD870 IS	30
40	Canon\tCanon PowerShot SX10 IS	5
41	Canon\tCanon PowerShot SX20 IS	3
42	CASIO\tQV-3000EX	2
43	CASIO COMPUTER CO.,LTD.\tEX-Z1000	4
44	CASIO COMPUTER CO.,LTD.\tEX-Z120	4
45	Cosina\tCosina CT-1A	1
46	EASTMAN KODAK COMPANY\tKODAK Z650 ZOOM DIGITAL CAMERA	5
47	FUJI PHOTO FILM CO., LTD.\tSP-2000	4
48	FUJIFILM\tFinePix Z1	3
49	KONICA MINOLTA\tDiMAGE Z3	1
50	LEICA\tD-LUX 4	2
51	LEICA\tD-LUX 5	6
52	motorola\tNexus 6	1
53	NIKON\tCOOLPIX P5000	3
54	NIKON\tCOOLPIX S210	1
55	NIKON\tCOOLPIX S4	3
56	NIKON\tE775	1
57	NIKON\tE7900	11
58	NIKON CORPORATION\tNIKON D200	31
59	NIKON CORPORATION\tNIKON D3	2
60	NIKON CORPORATION\tNIKON D300	2
61	NIKON CORPORATION\tNIKON D3100	1
62	NIKON CORPORATION\tNIKON D40	2
63	NIKON CORPORATION\tNIKON D40X	6
64	NIKON CORPORATION\tNIKON D50	2
65	NIKON CORPORATION\tNIKON D5000	2
66	NIKON CORPORATION\tNIKON D600	1
67	NIKON CORPORATION\tNIKON D70	1
68	NIKON CORPORATION\tNIKON D700	11
69	NIKON CORPORATION\tNIKON D70s	2
70	NIKON CORPORATION\tNIKON D7100	5
71	NIKON CORPORATION\tNIKON D80	19
72	NIKON CORPORATION\tNIKON D90	1
73	NORITSU KOKI\tEZ Controller	1
74	OLYMPUS CORPORATION\tu30D,S410D,u410D	1
75	OLYMPUS IMAGING CORP.\tE-M5	3
76	OLYMPUS IMAGING CORP.\tE-P1	5
77	OLYMPUS IMAGING CORP.\tFE180/X745	8
78	OLYMPUS IMAGING CORP.\tFE310,X840,C530	1
79	OLYMPUS IMAGING CORP.\tXZ-1	1
80	OLYMPUS OPTICAL CO.,LTD\tC4100Z,C4000Z	2
81	Panasonic\tDMC-LX1	4
82	Panasonic\tDMC-LX2	2
83	PENTAX\tPENTAX K-x	1
84	SAMSUNG\tSGH-I897	1
85	SAMSUNG TECHWIN\tVLUU NV 7, NV 7	1
86	SONY\tCYBERSHOT	1
87	SONY\tDSC-HX5V	1
88	SONY\tDSC-W150	8
89	SONY\tDSLR-A100	8
90	SONY\tNEX-3	2
91	Vignette\tVignette for Android	5

Appendix E: MeetUp API Integration

The MeetUp API provides a streaming interface to extract events happening in different cities by category as well as other types of data such as photos, community member profiles, notifications, topics, venues, rsvps, and other endpoints. Comparing the cities of Boston and Cambridge by category, "tech," "career/business," and "socializing" appear as the top three results for people to gather together. The other topics that draw a crowd include "health/well-being," "language/ethnic identity," and "outdoors/adventure" in order of popularity. The meetup categories may differ from city to city, but for Cambridge and Boston, these activities draw communities together. For Cambridge, Kendall Square is the technology hub and where the majority of meetup activity happens.

X.Category.	X.Boston.	X.Cambridge.	X.Cambridge.Perc.	X.Boston.Perc.
""	3	4	0.14	0.11
"alternative lifestyle"	16	16	0.57	0.57
"book clubs"	36	38	1.35	1.29
"career/business"	343	346	12.31	12.31
"cars/motorcycles"	14	15	0.53	0.50
"community/environment"	81	82	2.92	2.91
"dancing"	33	35	1.25	1.18
"education/learning"	74	75	2.67	2.66
"fashion/beauty"	11	11	0.39	0.39
"fine arts/culture"	59	58	2.06	2.12
"fitness"	80	79	2.81	2.87
"food/drink"	66	66	2.35	2.37
"games"	59	62	2.21	2.12
"health/wellbeing"	162	162	5.76	5.81
"hobbies/crafts"	33	33	1.17	1.18
ɹnguage/ethnic identity"	106	107	3.81	3.80
"LGBT"	62	61	2.17	2.23
"movements/politics"	37	37	1.32	1.33
"movies/film"	30	28	1.00	1.08
"music"	87	89	3.17	3.12
"new age/spirituality"	90	90	3.20	3.23
"outdoors/adventure"	102	102	3.63	3.66
"paranormal"	2	2	0.07	0.07
"parents/family"	93	91	3.24	3.34
"pets/animals"	36	34	1.21	1.29
"photography"	32	34	1.21	1.15
"religion/beliefs"	55	56	1.99	1.97
"sci-fi/fantasy"	19	20	0.71	0.68
"singles"	51	53	1.89	1.83
"socializing"	218	219	7.79	7.82
"sports/recreation"	91	91	3.24	3.27
"support"	69	67	2.38	2.48
"tech"	515	528	18.78	18.49
"writing"	21	20	0.71	0.75

MeetUp data extraction by city, category, and the percentage of popularity based on the category within a city.

Bibliography

Arfken, George B., Hans J. Weber, et al. Mathematical Models for Physicists: A Comprehensive Guide. 7th edition. Boston: Elsevier, 2013.

As, Imdat and Takehiko Nagakura. "Architecture for the Crowd by the Crowd": Architectural Practice in the Digital Age, ArchDesign Conference, 2016, Istanbul, Turkey.

Asay, Matt. "Open-source vs. Proprietary Software Bugs: Which Get Squashed Fastest?" *CNET*. CNET, 27 Sept. 2007. Web. 22 Mar. 2016. http://www.cnet.com/news/open-source-vs-proprietary-software-bugs-which-get-squashed-fastest/

Babak Saleh, Kanako Abe, et al. "Toward Automated Discovery of Artistic Influence." Computer Vision and Pattern Recognition. Cornell University. Vol 1408.3218, 2014.

Baughman, A., Gao, J. et.all (Eds.). Multimedia Data Mining and Analytics, Springer. 2015.

Bingham, N.H. Regression: Linear Models in Statistics. London, New York: Springer 2010.

Bomfim, Diney. "Cameras On Opengl ES 2.X - The Modelviewprojection Matrix".*Blog.db-in.com*. N. p., 2014.. 22 Sept. 2016. http://blog.db-in.com/cameras-on-opengl-es-2-x/

Bisgaard, Soren. Time Series Analysis and Forecasting by Example. Hoboken, N.J.: Wiley, 2011.

Bivand S. Roger, et. all. Applied Spatial Data Analysis with R. Springer, New York: 2015.

Brown, Nancy Clark. "BIM Benefits for Contractors ConAppGuru." *ConAppGuru*. Assemble Systems, 16 Nov. 2015. Web. 19 Feb. 2016.

Cowpertwait, S.P. Paul and Andrew Metcalfe. Introductory Time Series with R. Springer: New York. 2009.

Crotty, Ray. The Impact of Building Information Modeling: Transforming Construction. Abingdon, Oxon; New York: Spon, 2012.

Csurka, Gabriela, Chris Dance, et al. "Visual Categorization with Bag of Keypoints." ECCV International Workshop On Statistical Learning in Computer Vision, Prague, 2004.

Davenport, Thomas H, Jeanne G. Harris, et al. Analytics at Work: Smarter Decisions, Better Results. Boston, Mass: Harvard Business Press, 2010.

Davidson-Pilon, Cameron. Bayesian methods for hackers : probabilistic programming and bayesian inference. New York : Addison-Wesley, 2016

Davies, E.R. Computer and Machine Vision: Theory, Algorithms, Practicalities. Fourth edition. Waltham, Mass: Academic Press, 2012.

Ding, Xiaowen, Bing Liu, and Philip S Yu. "A holistic lexicon-based approach to opinion mining." *Proceedings of the 2008 International Conference on Web Search and Data Mining* 11 Feb. 2008: 231-240.

Forsyth, David. Computer Vision: A Modern Approach. Boston: Pearson, 2012.

Franses, Philip, Dick van Dijk, and Anne Opschoor. Time Series Models for Business and Economic Forecasting. Cambridge: Cambridge University Press, 2014.

Getis, Arthur and Barry Boots. Models of Spatial Processes: An Approach to the Study of Point, Line, and Area Patterns. Cambridge: Cambridge University Press, 1978.

Goldstein, Herbert, Charles P. Poole Jr., et al. Classical Mechanics (3rd edition). Essex: Pearson, 2014.

Gunelius, Susan. "The Cost Of Content Clutter Infographic". *ACI*. 2013. Web. 22 Sept. 2016. <http://aci.info/2013/11/06/the-cost-of-content-clutter-infographic/>

Guoshen Yu and Jean-Michael Morel, *ASIFT: An Algorithm for Fully Affine Invariant Comparison*, Image Processing On Line, 1 (2011).http://dx.doi.org/10.5201/ipol.2011.my-asift http://www.ipol.im/pub/art/2011/my-asift/

Haining, Robert P. Spatial Data Analysis: Theory and Practice. Cambridge; New York: Cambridge University Press, 2003.

Hansen, Per Christian. Least squares data fitting with applications, Víctor Pereyra, Godela Scherer. Baltimore, Md.:Johns Hopkins University Press, 2013.

Hartley, Richard and Andrew Zisserman. Multiple View Geometry in Computer Vision. Cambridge; New York: Cambridge University Press

Howe, Jeff. Crowdsourcing : why the power of the crowd is driving the future of business. New York: Three Rivers Press, 2009.

Kao Anne and Stephen R. Poteet (eds.). Natural Language Processing and Text Mining. London: Springer, 2007.

Mari, Dominique Drouet. Correlation and Dependence. London: Imperial College Press; River Edge, NJ: Distributed by World Scientific Pub. Co., 2001.

Matthews, John A. Quantitative and Statistical Approaches to Geography: A Practical Manual. Oxford; New York: Pergamon Press, 1981.

Mavinkurve, Ranjit, et al. "Improving Netflix's Operational Visibility with Real-Time Insight Tools." The Netflix Tech Blog. Jan 16, 2014. Accessed on June 16, 2015.

Miller, Harvey J., Jiawei Han. (eds). Geographic Data Mining and Knowledge Discovery. Boca Raton, FL: CRC Press, 2009.

Nagakura, Takehiko, Daniel Tsai, and Joshua Choi. "Capturing History Bit by Bit - Architectural Database of Photogrammetric Model and Panoramic Video." Proceedings of the 33rd eCAADe Conference - Volume 1, Vienna University of Technology, Vienna, Austria, 16-18 September 2015, pp. 685-694.
http://papers.cumincad.org/cgi-bin/works/BrowseAZ&name=authors/Show?ecaade2015_110

Negroponte, Nicholas. "Toward a Theory of Architecture Machines." *Journal of Architectural Education (1947-1974)* 23.2 (1969): 9-12.

Pfaff, Bernhard. Analysis of Integrated and Cointegrated Time Series with R: with 19 Figures. New York: Springer Science+Business Media, 2006.

Plant, Richard E. Spatial data analysis in ecology and agriculture using R. Boca Raton : CRC Press, c2012.

Quirk, Vanessa. "A Brief History of BIM / Michael S. Bergin | ArchDaily." 2012. 6 Jun. 2013

Rice, J.A. Mathematical Statistics and Data Analysis, 3rd ed. Thomson, Belmont: 2007.

Ripley, Brian D. Spatial Statistics. Hoboken, NJ: Wiley-Interscience, 2004.

Rogerson, Peter. Statistical Methods for Geography: A Student Guide. London; Thousands Oaks, CA: SAGE, 2006.

Simon, Phil. "Big Data Lessons from Netflix." Wired. March 2014
http://www.wired.com/2014/03/big-data-lessons-netflix/

Snavely, Noah, Steven Seitz, and Richard Szeliski. "Modeling the World from Internet Photo Collections." International Journal of Computer Vision. Volume 80 Issue 2, Nov. 2008. Pages 189-210. http://phototour.cs.washington.edu/ModelingTheWorld_ijcv07.pdf

Song, Peter. Correlated Data Analysis: Modeling, Analytics, and Applications. New York: Springer, 2007.

Sumathi, S. and Sivanandam, S.N. Introduction to Data Mining and its Applications (Studies in Computational Intelligence). 1st edn. Springer. 2006.

Surowiecki, James. The Wisdom of Crowds. New York: Anchor Books, 2005.

Tadisco, Patrice. *Copley Square*. (2012). *Landscape Notes*. Retrieved 22 September 2016, <https://landscapenotes.com/2012/08/23/copley-square/>

Tomasik Brian, Phyo Thiha, and Douglas Turnbull. Image from "Tagging Products using Image Classification" , SIGIR 2009.

Trucco, Emanuele. Introductory Technique for 3-D Computer Vision. Upper Saddle River, NJ: Prentice Hall, 1998.

Van Leeuwen Jos P., Harry J Timmermans. Innovations in Design and Decision Support Systems in Architecture and Urban Planning. Dordrecht: Springer, 2006.

Vanderzyle, Bruce. "Business Intelligence in the Construction Industry". Presented at the Chapter of the Construction Financial Management Association (CFMA) of Massachusetts.

Wardel, Jane. "Singapore Airlines Orders Airbus Planes". The Washington Post. July 21,2006 http://www.washingtonpost.com/wp-dyn/content/article/2006/07/21/AR2006072100744.html

Weiss M. Sholom...[et al]. Text Mining: Predictive Methods for Analyzing Unstructured Information. New York: Springer, 2005.

Zanasi, A., C.A. Brebbia, et al. (eds) International Conference on Data Mining (8th:2007: Data Mining VIII: Data, Text, and Web Mining and their Business Applications. Southampton: WIT Press, 2007.

Correlation and dependence. In *Wikipedia.* 25 December 2015. https://en.wikipedia.org/wiki/Correlation_and_dependence

The Eiffel Tower: official website. 2016. Web. Accessed June 9, 2016. http://www.toureiffel.paris/en/everything-about-the-tower/themed-files/71.html.

"Andy Bromberg." *Sentiment Analysis*. Web. 13 May 2014.

"How The USGS Uses Twitter Data To Track Earthquakes | Twitter Blogs".*Blog.twitter.com*. 2015. Web. 22 Sept. 2016. <https://blog.twitter.com/2015/usgs-twitter-data-earthquake-detection>

CEFS HOME. Eco.ucdavis.edu. Retrieved 10 August 2016, http://eco.ucdavis.edu/

"When A Machine Learning Algorithm Studied Fine Art Paintings, It Saw Things Art Historians Had Never Noticed". The Physics arXiv blog. Accessed 23 August 2015.

SIFT - OpenCV documentation
http://docs.opencv.org/trunk/doc/py_tutorials/py_feature2d/py_sift_intro/py_sift_intro.html

Business Intelligence In The Construction Industry - Massachusetts." *Cafe.cfma.org*. N. p., 2016. Web. 22 Sept. 2016. http://cafe.cfma.org/MassBostonMA/resources/resources/viewdocument/?DocumentKey=885d079c-eceb-41af-9983-b7952182b87b

"Boston's People and Economy." Vol. 1, 2009. Published by the City of Boston. http://www.cityofboston.gov/

"Correlation coefficient" by Kiatdd - Own work. Licensed under CC BY-SA 3.0 via Commons - https://commons.wikimedia.org/wiki/File:Correlation_coefficient.png#/media/File:Correlation_coefficient

"Point Cloud". *Wikipedia*. N. p., 2016. Web. 22 Sept. 2016. https://en.wikipedia.org/wiki/Point_cloud

Giovanni Battista Piranesi - R.S. Johnson fine art, Public Domain, https://commons.wikimedia.org/w/index.php?curid=5765775