# Will They Buy?

by

Rony Daniel Kubat

S.B. Computer Science, MIT 2001
S.B. Mechanical Engineering, MIT 2001
S.M. Computer Science, MIT 2008

Submitted to the Department of Electrical Engineering
and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Computer Science and Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2012



Author .....
Department of Electrical Engineering
and Computer Science
January 1, 2012

Certified by .....
Deb K. Roy
Associate Professor
Thesis Supervisor

Accepted by .....
Professor Leslie A. Kolodziejski
Chairman, Department Committee on Graduate Students

**Will They Buy?**
by
Rony Daniel Kubat

Submitted to the Department of Electrical Engineering
and Computer Science
on January 1, 2012, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Computer Science and Engineering

## Abstract

The proliferation of inexpensive video recording hardware and enormous storage capacity has enabled the collection of retail customer behavior at an unprecedented scale. The vast majority of this data is used for theft prevention and never used to better understand the customer. In what ways can this huge corpus be leveraged to improve the experience of customer and the performance of the store?

This thesis presents MIMIC, a system that processes video captured in a retail store into predictions about customer proclivity to purchase. MIMIC relies on the observation that aggregate patterns of all of a store's patrons—the gestalt—captures behavior indicative of an imminent transaction. Video is distilled into a homogenous feature vector that captures the activity distribution by first tracking the locations of customers, then discretizing their movements into a feature vector using a collection of *functional locations*—areas of the store relevant to the tasks of patrons and employees. A time series of these feature vectors can then be classified as predictive-of-transaction using a Hidden Markov Model.

MIMIC is evaluated on a small operational retail store located in the Mall of America near Minneapolis, Minnesota. Its performance is characterized across a wide cross-section of the model's parameters. Through manipulation of the training data supplied to MIMIC, the behavior of customers in the store can be examined at fine levels of detail without foregoing the potential afforded by big data.

MIMIC enables a suite of valuable tools. For ethnographic researchers, it offers a technique for identifying key moments in hundreds or thousands of hours of raw video. Retail managers gain a fine-grained metric to evaluate the performance of their stores, and interior designers acquire a critical component in a store layout optimization framework.

Thesis Supervisor: Deb K. Roy
Title: Associate Professor

## Acknowledgments

This thesis would never have existed without the support and encouragement of my advisor, Deb Roy. I'm grateful that he agreed to take a chance with this out-of-department and naïve twenty-six year-old, six years ago. Deb's one to throw opportunities at his students—one after another, the kind of opportunities to be involved with bold projects that many graduate students can only dream. I feel very lucky to have called our research group home for the last six years.

My committee, Professors Eric Grimson, and Leslie Kaelbling, have also been a great support for my efforts; they've provided insightful feedback, and helped rein in the scope of a thesis that could easily have gotten out of hand.

Several of my fellow cogmac graduate students need to be singled out. Brandon Roy, my officemate during the last three years, has been a great source of camaraderie and encouragement. He's always been willing to pour in his insight and time in the many technical challenges I faced during the course of this work. My thesis would not have been possible without Philip DeCamp. If our intellectual growth and research stands on the shoulders of giants, my code stands on the shoulders of Phil. I have learned more about the art of developing software from Phil than in my five years as an undergraduate in computer science. The extensive infrastructure on which this thesis is built is largely Phil's doing.

My former officemates Stefanie Tellex and Michael Fleischman and fellow cogmac'er Kai-yuh Hsiao have been fonts of encouragement, wisdom and great discussion. The graduate life sometimes feels like an endless tunnel, and these three doctors showed me that the pinprick of light at the other end was not an optical illusion, but in reality much closer than I imagined. Matthew Miller and George Shaw have been great collaborators on this project; they created excellent building blocks from which to construct this work, and have been a constant source of fascinating discussions—about the sciences or about life. Cogmac'ers Jeff Orkin and Soroush Vosoughi have helped make our research group awesome.

Karina Lundahl has been a miracle worker and great friend; she's created an atmosphere in our lab that makes it a joy to arrive in the office each day. Always with a sympathetic ear and a good story, Karina's tethered our research group to a humanistic earth—she's provided a grounding to my intellectual hot-air balloon.

It goes without saying that this work would have been impossible without the support and encouragement of the corporate members of the Media Lab.

Ken Jackowitz, at Bank of America, was the first to recognize the potential of the Speechome recordings for use in retail. He has been wonderful—one of the visionaries in the corporate world who understands the value of the speculative research of an academic institution such as MIT. Neil McPhail of Best Buy also understands our vision, and made possible this collaboration. Matt Skally flew the Media Lab flag within Best Buy; thanks! Eric Pedersen has been invaluable; he's been a wrangler of data, an advocate for our work, and always patient and generous.

A special thanks to Katja Schechtner who provided insightful advice and support at several critical junctures of this research.

My early mentors put me on this path. Seth Weiss, James McLurkin, Rod Brooks: thanks! From those very early days, Professor Alex Slocum's kind words had lasting influence, spurning me toward a graduate degree. Working in Deb's lab would have been impossible were it not for the direct support of the National Science Foundation's Graduate Research Fellowship. I am greatly indebted to the NSF, my reviewers and those who recommended me.

Outside and inside the institute, I have been supported by an incredible network of close friends. Dan, Mark, Joe, Ania, Aude, Angelica, Stephen, Luke, Steve, /, ln, La famille Lac, Natan, Edwina, Jeremy, Kat, Emily, Michael, Alan, Janet, Chao, your trust and friendship means the world to me.

Finally, there can never be enough thanks for my parents, Doctors Hadassa and Peter Kubat, who instilled in me a curiosity, who always encouraged me to pursue my dreams, wherever they lead, and who have given be unwavering, unconditional support and love. This work is dedicated to you both.

# Contents

# List of Figures

# List of Tables

Chapter 1

# Introverview

When was the last time you went to the store? Was it over the weekend, investigating a new washing machine? Or last night, visiting the supermarket to pick up fresh tomatoes and milk? This morning buying a watch battery? Retail shopping is unavoidable, an integral part of the American experience. And if you are like the average American, over the next year, you will spend nearly ten thousand dollars in retail stores. Retail is an enormous driver of the economy. Americans spend in excess of three *trillion* dollars each year in retail, roughly 20% of the national gross domestic product. The total sales less the cost of goods sold—the gross margin—tops eight-hundred billion dollars (US Census Bureau, 2011a,c,b).

With such enormous stakes, retailers have spent enormous effort to understand the behavior of the consumer. They have analyzed reams of transactional footprints, examined surveys and focus groups. They viewed thousands of hours of video through the lens of an ethnographer. An entire industry is devoted to consulting on customer behavior, and providing tools to capture various quantitative measures of customer behavior and preferences.

The last twenty years have seen tremendous technological change that has multiplied by orders of magnitude the data available to retailers: inexpensive video recording devices and immense data storage warehouses have enabled the collection of behavioral data at unprecedented scale. Till now, this data has been collected primarily for loss prevention. Once recorded, this video data is left untouched, never to be reviewed. To gain insights about customer behavior—the patterns of behavior which inform a customer's decisions—several questions are prompted: Who will watch the millions of hours of recorded video? How can any manual process extract meaningful

insights from such a data deluge? This thesis takes advantage of inexpensive computational processing power to automate some of the tedious customer-watching tasks and provide a set of tools that can help characterize behavior, find episodes worthy of detailed examination, and, potentially, help evaluate and optimize the physical layout of stores. The following chapters present this computational tool for quantitatively understanding human behavior in stores, an end-to-end system that uses patterns of movement captured on video to make predictions about customer purchases. Ultimately, the hope is to create a better experience for the customer and to increase the efficiency of retail stores, driving down the cost for consumers and raising the profits of the retailer.

## 1.1 Consumer Behavior in a Brick-and-mortar Store

The most admired company from a retail perspective is Apple Computer. Last year, Apple's retail sales per square foot exceed all other major retailers by a huge margin, nearly twice the second-place finisher, Tiffany & Co. (Retail Sails, 2011). Why are Apple stores so productive? Certainly, Apple's products engender strong emotional reactions from their customers—an emotional connection that ultimately manifests in sales. Apple's stores are productive because there is strong demand for the products within, but walk into an Apple store, and it is hard not to notice a difference compared to other stores. Apple's retail environments are inviting, with a clean aesthetic that matches their products and a spacious, uncluttered layout of tables, benches and displays. If we wanted to divide an Apple store's performance into the individual influences of product, employee, lighting, aesthetics, layout—among a wide variety of other factors—how can we separate these intertwined factors? Retail stores are never isolated. Their performance is influenced by factors far outside the building's four walls. The visceral reaction to a brand of car is cultivated long before a buyer sets foot in the showroom floor. This is the rabbit-hole of complications studying the retail environment.

Taking a broader view, a retail store is an instance of the class of narrowly goal-directed spaces. Goal-directed, in this context, means that the ultimate users of the space—the customers in the retail store—are motivated to accomplish one or more of a small set of tasks.[1] Other examples of these goal-directed spaces include public transit hubs, libraries, medical offices and factories. We can contrast this with spaces without such regularity of

[1] Of course, the employees also have distinct goals in their use of the space. Here, I focus on the consumer.

**Figure 1.1:** The Apple retail store in the upper west side, New York City. (Images: © Apple)

purpose. Homes and office buildings have enormously greater variety in the activities that take place within.

The smaller set of expected behaviors in goal-directed spaces gives designers, analysts and managers a better way to evaluate the space. These spaces have clearer objective measures of performance than their more multifaceted cousins. Herein lies two important consequences. First, a restricted set of behaviors implies an easier task of modeling macroscopic phenomena—all things being equal, a model of fewer behaviors will have greater predictive power. Second, if one's goals are to optimize the performance of the space, the objective measures afforded by highly goal-directed spaces are crucial. They give guidance whether one space is "better" or "worse" than another. The multitude of competing interests in more general-purpose spaces make comparison between candidate spaces a more subjective judgement.

Returning our view to the retail space, some objective measures in stores are clear. A store's profit margin is objective, important and easily measured; ultimately, it is the final arbiter of success or failure. Other measures of concern to retailers are not as clear. A customer's relationship with the brand, store or employees is subjective. These emotional consequences of the contact of customer and store are hard to measure without an exit survey or other explicit intervention. Another challenging confound is the interaction between the retail experience and other sales channels. If you need to buy a washing machine, you might visit the local Sears to learn about the technologies and different products available—to kick the tires, so to speak—then choose to buy while sitting on your living-room couch,

making the purchase through Sears online or other internet retailer.[2]

Consumer behavior in a retail store—or the behavior of occupants in goal-directed spaces—is an enormously broad and rich field. This thesis examines one relevant and constrained corner of the discipline.

## 1.2 The Scope

Even in a restricted retail setting such as a boutique chocolatier, a framework for understanding the behavior of the consumer is an enormous undertaking. Why is the customer there? What motivated her entry into the store? What factors determine the length of her stay, the size of her purchase (if she purchases at all)? Do product samples decrease the purchase "activation energy"? What behavior changes when the purchase is for personal consumption versus a gift? How does the location of the store in the neighborhood affect the quantity and frequency of customer visits? What effect color? Smell? Light intensity? The texture of the materials of product packaging? One should not forget the consequences of a customer's demographic: gender, age, economic-segment, culture. A list of these influences and potential influences could continue ad infinitum, but importantly, these influences are intertwined and inseparable through simple models.

This thesis frames a narrow but very important sector of retail consumer behavior. Herein, I develop a model for small retail spaces on the order of several thousand square feet, typical in size for the small stores that fill the central portions of the traditional American mall. As an objective measure of performance, this model assesses the final arbiter of a store's success: sales. The bases for the model are the physical patterns of movements of customers within the store. Of the knotted bundle of influences, the one I try to disentangle is the gross physicality of customer movement. More specifically, this thesis pivots on a central hypothesis that there are qualities of patterns in the comprehensive *distribution* of people in the store that are predictive of sales. Essentially: what other customers are doing in a store correlates with your propensity to make a purchase. Stated succinctly:

> The occurrence of a transaction can be predicted from the temporal *patterns* of activity distributions in the entire store. That is to say, there are measurable differences between the distribution

[2] Though internet commerce has radically shaped the landscape of sales, this cross-channel confound has long history. Take for example this story of a Sears retail store. For decades, the printed catalogs of Sears drove a significant fraction of their revenue.

of customers in a store preceding a sale and the distribution of customers when no sale occurs.

The validation of this hypothesis will empower retailers with a suite of tools to measure and improve their stores, an outcome that will ultimately help the consumer as well. The validation comes through the demonstration of an end-to-end system that successfully predicts customer transactions in an operational store.

The demonstration begins with raw video data captured using overhead cameras in a store. The video is processed in several steps to distill customer motion which is then fed to a classification model which discriminates patterns of customer motion indicative of purchase from those which are not. Each step along the processing pipeline employs techniques from computer science—chiefly computer vision and machine learning. The aim is not to advance individual processing components, per se. Rather, the objective of this holistic system is to demonstrate-by-example the validity of the central hypothesis and inaugurate the consequent suite of tools. The goal is to build a *system*. Beyond the immediate learnings and tools this system affords, this thesis is an arrow pointing to future paths. One can have a sensible expectation that performance of the system will be improved by substituting best-in-class algorithms for components in the framework.

## 1.3 The Challenge

Even with the reduced scope of patterns of activity in small retail stores, there are two deep practical challenges building the system. First is the inherent difficulty of unwinding the influence of activity patterns from the multitude of confounds, especially in our case where we have data from only one store. Second are the technical hurdles that must be overcome to automate the data collection and processing.

In at least one extreme, activity patterns consciously affect purchasing behavior: Faced with a long line at the check-out counter, customers will sometimes choose to abandon their shopping and depart empty-handed. But in most other circumstances, we might expect the influence of these patterns of activity to be subtle and subconscious. The signal, in effect, is swamped in a sea of noise: the many other influences on behavior. The type of product being purchased also has a huge impact on behavior. Some, like chewing gum, are impulse-buys. Others (an engagement ring) may demand long deliberation and negotiation. The cell phone store modeled in this thesis has a spread of both types of products.

A panoply of technical challenges bedevil the customer modeling pipeline. Take as example the problem of tracking the location of customers and employees in the store using overhead cameras. Occlusions of customers with each other and store furnishings are frequent. The overhead camera's perspective on a person changes radically due to the extreme wide-angle lenses used. The frequently-seen behavior of customers standing still for minutes examining a product wreaks havoc on a tracker's foreground/background segmentation. Merging the resulting trajectories from multiple cameras into a single coherent path is fraught with difficulty.

Given all these challenges, an end-to-end system suffers an additional burden. In a system such as ours which connects multiple complicated parts, errors propagate and expand, diminishing further the signal we are attempting to capture.

The challenges this thesis confronts are formidable, but the potential consequences of success are immense.

## 1.4 The Value of Prediction

Assuming success building a mechanism that can give forewarning of a purchase, how can retailers leverage the model to improve the experience of the customer and reduce the cost of selling? I propose three tools.

The first tool aids researchers. Most of the millions of hours of video collected yearly in retail stores are never seen by human eyes. A successful model of purchasing behavior can help identify episodes when, with high probability, there *should* have been a transaction, but none took place. Bringing trained human eyes to these "near misses" may identify systemic opportunities for improvement in staff training or scheduling, the layout of the store or other retail elements.

Secondly, by bringing real-time or near real-time updates to a store or regional manager, a prediction tool makes retailers more agile. Finally, when coupled with a generative model of customer movement, the model can be used to optimize the physical layout of a store. The ultimate outcome of each of these tools is to reduce a customer's frustration, and increase their satisfaction.

## 1.5 The Strategy of this Thesis

A fifty-thousand-foot overview of the machinery developed in this thesis—a system I dub Mimic—will help guide the detailed explanation of the

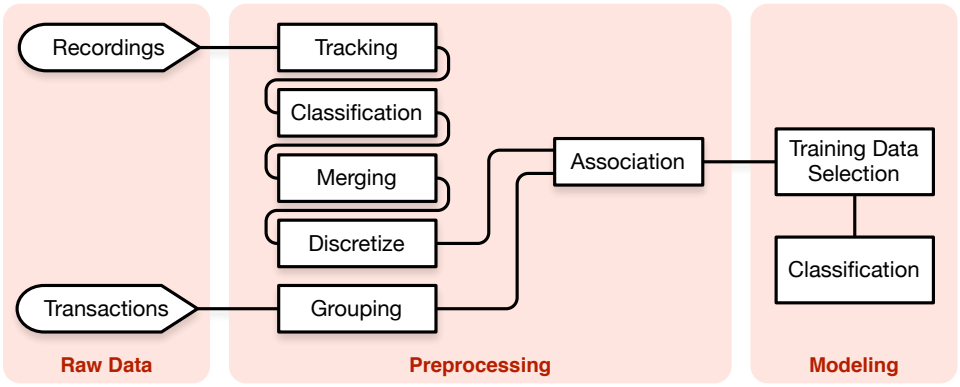


**Figure 1.2:** The data pipeline of the MIMIC transaction classifier. Raw data consists of multiple video streams from a store and the timestamped transactional record. Pedestrians are tracked in the video stream and classified as employees or customers. Next, trajectories from the multiple cameras are combined then discretized into a feature vector based on functionally-relevant locations in the store. Finally, an HMM-based model classifies spans of time as either preceding-transaction or not.

subsequent chapters. A schematic of the flow of data through the steps below is shown in Figure 1.2.

- We begin with raw data in the form of video captured from several overhead cameras mounted near the ceiling of the store (A still frame from one of these cameras shown in Figure 1.3). The raw video then goes through a series of preprocessing steps to extract customer behavior patterns and pack them into a form that can be modeled using standard machine-learning techniques.

- Next, video is spatially subsampled to reduce the computational requirements of subsequent steps. Next, people in each of the video streams are tracked and classified as either customer or employee. The tracking module generates these trajectories in image coordinates which must then be translated into a global Euclidean space. Trajectories generated from different cameras are then merged due to overlap in the camera's fields of view. Several cameras track the same person at the same time, and as a person moves from one part of the store to another, the associated trajectory must be handed off between several cameras.

- Central to this thesis is the concept of *functional locations*, or *flocs*. These are volumes surrounding areas of the store that may have rele-


**Figure 1.3:** A typical frame of video from an in-store overhead camera. This frame is from the Best Buy Mobile case study described in Chapter 4. Video was recorded at 960 by 960 resolution at approximately fourteen frames per second.

vance to the various goals or tasks of customers in the store. Flocs are primarily used to quantify the patterns of customer activity in the store by discretizing very high-dimensional trajectories output by the tracker into a homogenized sequence of feature vectors. This sequence of feature vectors discards potentially relevant information, especially the individual trajectories of customers, but also mitigates the propagation of errors present in the tracking data.

- An electronic transaction record (i.e. the receipts of store sales) serves as a secondary input to MIMIC in addition to the raw video. The timestamped records identify training data in the stream of activity pattern feature vectors, dividing it into fixed duration episodes labeled as preceding-transaction or not.

- The final stage of MIMIC is a classifier comprising hidden Markov models trained on the labeled episodes.

This holistic system is used to validate the core hypothesis—that patterns of activity are predictive of transactions—using data collected from an operating store. MIMIC's ability to successfully predict transactions serves as a proof-by-example that there exists a predictive signal present in the aggregated dynamic patterns of customer distributions in a store. By inspecting

the models learned in the final stage of the system and by exploring the learned model on variations of activity data, we can learn about customer behaviors in the store.

## 1.6 Roadmap

The structure of this document follows:

**Chapter 2: Context and Related Work** This chapter gives background to the many fields touched by this thesis, outlining previous research and commercial solutions in the domain.

**Chapter 3: Mimic: Prediction using Functional Locations** Next, I introduce *functional locations* as a low-dimensional feature useful for characterizing the patterns of activity within a retail store, and Mimic, a model I developed to predict transactions from the low-dimensional patterns of activity observed in a retail store.

**Chapter 4: A Case Study** We teamed with Best Buy to install a video capture system within one of Best Buy's smaller stores in the Mall of America near Minneapolis, Minnesota. This chapter details the design and implementation of a system that captures real-world activity pattern data from a retail store—fodder for evaluating the Mimic model.

**Chapter 5: Performance of the Mimic model** Here, I examine the performance of Mimic on the corpus of data collected in our case study, explore the patterns of behavior captured in a model, and describe several experiments enabled by Mimic.

**Chapter 6: Conclusion** I end with a discussion of the tools that are enabled by Mimic, directions for future research and a summary of this the contributions of this thesis.

## A Note About Language

The work presented in this thesis stands on the shoulders of several collaborators in our research group, especially Philip DeCamp, Matthew Miller, and George Shaw. Where elements of this thesis result from collaborations,

I have made references to the individuals and any existing documentation of the work. Furthermore, I make distinction between my work and that completed in collaboration by shifting the subject in the writing: work completed alone uses the first-person singular, work completed in collaboration uses the first-person plural.

Chapter 2

# Context and Related Work

As an end-to-end system that predicts customer behavior from video, this thesis touches several different domains of study, each with long history worthy of a complete chapter. The goal here is to situate the thesis in context and reflect it against related approaches in each of the several disciplines rather than exhaustively survey each field of study. MIMIC, the model described and demonstrated in the subsequent chapters, tries to understand physical behavior of consumers in indoor environments.

In this chapter, I will be discussing models of human (and more specifically, customer) behavior in buildings, as well as computational tools modeling behavior in video.

The activity of shopping can be broken along many dimensions. There are sub-activities common among most shopping experiences: activities such as browsing, comparing, searching, and purchasing. There are subclasses of shopping events dependent on what and how much is being purchased—compare, for example, grocery-shopping to jewelry-shopping. Shopping behaviors overlap within a single shopping episode as when a customer's directed search for a staple such as milk is displaced by a tempting ancillary item discovered during the search.

Detailed analysis of consumer behavior falls under the purview of sociologists, ethnographers, and most specifically, consumer marketing researchers. These researchers draw many of their conclusions from intensive human observation, a methodology that has the advantage of bringing all a researcher's cognitive power to understand the subtleties of human behavior. In contrast, some researchers studying behavior in public spaces (space-syntacticians, for example) bring a computational and data-driven perspective—modeling

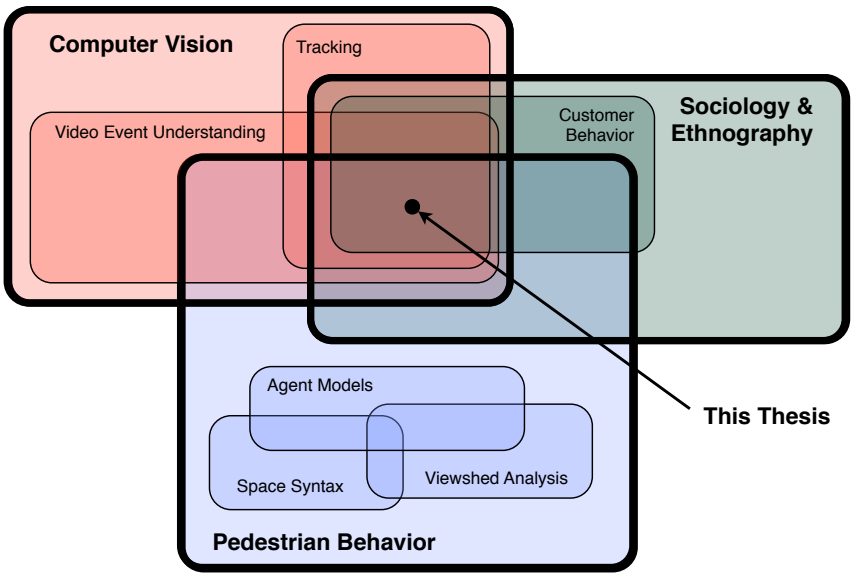


**Figure 2.1:** How this thesis fits in among its many related disciplines.

pedestrians using their trajectory through space, and correlating that behavior with measures derived from the shape and configuration of the space.

Another relevant connection for the Mimic model is computer vision; Mimic touches on several decades of research in computational vision—from low-level background/foreground segmentation to high-level event-understanding. Many of these techniques employ temporal models, of which Mimic's use of hidden Markov models (HMMs) is but one example.

The rest of this chapter examines these areas of study in greater detail and reflects on how the Mimic model and the case study described in Chapter 4 relate to them. Figure 2.1 places this thesis in context visually.

## 2.1 Sociology and Ethnography: Methodologies for Examining Customer Behavior

The methodological approach of many researchers interested in consumer behavior can best be exemplified by Erving Goffman's 1966 work, *Behavior in public places: Notes on the social organization of gatherings* (Goffman, 1966). Goffman developed a framework for social engagement—face-to-face interactions, social occasions, self-involvement, and the boundaries of social engagement—from personal observations made in a small farming

community, in a mental institution, and from generalizations drawn from etiquette manuals. Goffman is concerned with an individual's sense of self within a community; he scrutinizes facial expression, gesture, and stance within a social gathering, relating them to the formality of the moment (what he describes as "tightness and looseness") and the structure and allocation of a person's involvement. Goffman's interpretations rely on his personal understanding of the human condition; his analysis of behavior comes from an ability to place himself "into another's shoes" and imagine the conscious and unconscious desires and obstacles present in those participating in social engagement. The observation methodology exemplified by Goffman is used by sociologists and ethnographers interested in consumer behavior in retail environments.

Paco Underhill's influential book, *Why we buy: The science of shopping*, brings this approach to consumer behavior in retail settings (Underhill, 2000). Underhill directs a consultancy focused on customer behavior and draws conclusions from the years of detailed ethnographic studies his consultancy performed in stores. His employees followed customers during shopping trips, diligently recording actions taken.[1] He writes:

> In addition to measuring and counting every significant motion of a single shopping trip, our trackers also have to contribute incisive field notes describing the nuances of customer behavior and make good inferences based on what they've learned. These notes add up to yet another, this time anecdotal, layer of information about a particular environment and how people use it. (p. 7)

The great advantage of the observational methodologies used by ethnographers and sociologists is that researchers bring their personal understanding of the human condition. Most of Underhill's inferences and recommendations are derived from this anecdotal layer of information. Several examples: The "decompression zone" at the entrance of a store moderating a customer's first moments in the store, "hand-allotment" and the availability of shopping baskets, and the effects of signage and seating location. Take as another example the "butt-brush effect" described by Underhill: in clothing stores, when clothing racks are so closely packed that a customer makes contact with

[1] It is likely that Underhill would not think too highly of the method applied and presented in this thesis. He writes in *Why We Buy*: "Some of the stuff I get is outright silly, like a software package designed to track tank movements from spy satellites. Put enough cameras with wide-angle lenses into your ceiling and voilà!—instant science of shopping." (p. 27)

objects or people behind them, they are likely to move away. The discomfort of being touched, he argues, subtly suppresses the browsing impulse. That simple observation—and its explanation in particular—embeds an understanding of the human experience that is unlikely to be easily encoded in a computational model.[2]

The human observers that ethnographers employ are also the discipline's disadvantage: human observers are expensive, both in time and money. Observations are costly to collect and to code; they are unscalable beyond short studies in a handful of locations. Regardless of precise coding methods, data is influenced by the subjectivity of the person doing the observing.

In contrast, some quantitative measures can be gathered automatically. Underhill describes criteria retailers can use to quantify the performance of their stores in relation to the customer experience such as the commonly used *conversion rate* (the fraction of shoppers entering the store that make a purchase), time spent in the store, the *confusion index* (how navigable is a store), *interception rate* (fraction of shoppers who make contact with an employee) and waiting time. Many of these measures can be collected automatically and there is a burgeoning industry of commercial solutions which provide such data inexpensively via cameras or counters. No doubt, much retail customer behavior research remains proprietary; a better understanding of the customer gives a retailer competitive advantage. Underhill's book results from his experience consulting to large retailers, as does, for example, Herb Sorensen's *Inside the mind of the shopper*, and Martin Lindstrom's *Buy • ology* and *Brand Sense*.

The ethnographic stance of intensive human observation is extremely valuable in understanding consumer behavior, but its drawbacks of cost and efficiency prevent ubiquitous deployment as an instrument for retail. This presents an opportunity for automated tools to aid ethnographers and retail analysts—tools such as those presented in the following chapters. Next, we discuss data-driven approaches that capture some aspects of customer

[2] The subtlety of the solutions proposed by Underhill and others consulting in consumer behavior is reminiscent of a famous anecdote from Operations Research lore: Complaints of long wait times for elevators in a high-rise hotel led to an analysis by operations researchers as to how elevator queuing could be improved. The simple, human, solution: place mirrors near the elevators allowing those waiting to "fix their ties, comb their hair, and even perhaps coyly flirt via the mirror with others who are likewise waiting. . . [T]hose hotels that invested in such mirrors received far fewer complaints about elevator delays than competitors who did not." (Larson, 1987) This thesis enables the identification of cases when an expected purchase was not made. These can later be classified by human observers who may offer solutions (like the mirrors by elevators) that increase purchasing behavior.

behavior, analogues to this thesis' work.

## 2.2 Quantitative Approaches to Customer Behavior Analysis

In retail, the easiest data-trail to harness for customer behavior modeling is the transactional record made at the point-of-sale (POS). Since large-scale statistical programs became affordable, this data has been used to model behavior as diverse as brand choice (Guadagni and Little, 1983) and the impact of promotions (Gupta, 1988). Long-term customer patterns can be mined when transactions are associated with customer ID, for example, with loyalty cards (Hamuro et al., 2002; Yada et al., 2006). Though point-of-sale records can be used to align *what* was purchased with *where* in the store the products were retrieved, the path taken by customers to collect the products is lost. This path information can be enormously informative—illuminating the successes and failures of both product and venue. Online retailers do not suffer from this drawback.

Compared to the point-of-sale transactional record available to retailers with physical stores, online retailers have enormously more data to mine. When a customer browses the Amazon online store for a baby carriage, he leaves behind a valuable record of the route taken before a purchase is made. Amazon knows what other carriages were considered; how long the customer spent on each page before proceeding; they know when (and how) a user browsed for a baby carriage *even if a purchase was never completed*. This "trajectory" data is invaluable for providing quality recommendations to the consumer, pricing items appropriately, and understanding the customer at a much finer level of granularity than the strictly transactional techniques of retailers. Researchers have been mining patterns in web usage (Catledge and Pitkow, 1995; Srivastava et al., 2000; Borges and Levene, 2000; Montgomery et al., 2004; Moe, 2006) and automatically adapting to user browsing (Perkowitz and Etzioni, 1997; Anderson et al., 2002) since the early days of the world wide web.

Retail websites have an experimental advantage over brick-and-mortar retailers. Subtle changes to the wording of a promotion or the navigation of the site can be rapidly tested on subpopulations of customers; each iteration is fast and inexpensive.[3] Retail stores can neither change their physical

[3] This type of A/B testing is now very easy to apply to any sort of website. The last few years saw a crop of new companies specializing in tools to make testing easier, Visual Website Optimizer, SiteSpect, and Google Website Optimizer to name a few.

layout as quickly, nor as easily control the demographics of the customers in such experiments. One goal of this thesis is to provide a key component of a toolset which can be used to virtually test the physical configuration of stores, reducing the need for costly real-life experiments in stores.

One of the richest sources of quantitative data about customer behavior are the paths customers take through stores. Customer paths are highly constrained by a store's layout; and Herb Sorensen argues that eighty percent of the purchase decisions made by a customer are decided by a store's layout (Sorensen, 2009).[4] In a coarse way, retailers already act on this conjecture. Consider the familiar example of refrigerated staple goods like milk placed at the rear of the supermarket, forcing customers to walk past a panoply of purchase opportunities.

### 2.2.1 Where Customers Go: Analyzing Consumer's Trajectories

Farley and Ring's stochastic model of customer's paths was one of the earliest models of customers in a retail store (Farley and Ring, 1966). This model divided supermarkets into several large regions. Customers pass from one region to another with a probability derived and modeled using the *content* of the areas (modeled as a "force" connected to sales volume) as well as the characteristics of the store's configuration—for example, the observation that customers often circle the perimeter of the store in a counter-clockwise direction. Farley and Ring validated their model through direct observation in five Pittsburg supermarkets.

In the 1990s, RFID and WiFi technologies made the collection of large datasets of the paths consumers take within a store economical. One technology, PathTracker (Sorensen, 2003), developed by Herb Sorensen and Sorensen Associates has been used in several studies of customer paths. PathTracker locates RFID tags attached to shopping carts and baskets using a network of sensors at the periphery of a store. It has an accuracy of several feet, small enough that customer paths can be localized to a specific register and then associated with a transactional record. Using data from this system, Larson et al. clustered several thousand customer trajectories in a supermarket, normalizing their durations and using mean point-wise Euclidean distance as the path distance metric to a modified k-means clustering algorithm (Larson et al., 2005). The exploratory dataset confirmed conventional wisdom that

[4] Many of the examples Sorensen cites are in supermarkets; it's unclear if sales in other types of retail environments are also so highly dependent on layout.

the periphery of the store (the "racetrack") was utilized much more than aisle-ways.

Yada also used trajectories of supermarket customers gathered using RFID (Yada, 2011). Trajectories were discretized into a sequence of passages through labeled areas; decision trees were then used to classify customer paths as either "high-volume" or "low-volume". The manually annotated areas used in this classifier are similar to the functional locations described in the following chapter. In both cases, these areas are used to dramatically reduce the complexity of trajectory data. Unlike the functional locations used in Mimic, the supermarket's subdivided areas are very large, encompassing several aisles each. In a related supermarket study, Kholod et al. found strong correlation between a customer's total path length and the volume of his purchases (Kholod et al., 2010). In addition, they clustered customer behavior into three types (wandering, decisive and mixed) using the distribution of wandering-degree, a ratio between distance traveled and the area of shopping zone in which a shopper moves.

Hui et al. proposed several hypotheses of customer behavior and used supermarket customer trajectory data to validate their hypotheses (Hui et al., 2009a). In particular, they argued that customers become more purposeful—spending less time exploring and more focused on purchases—as they spend time in the store, that the presence of other customers in an area attracts visits yet repels purchases, and that shopping virtuous categories (e.g. health food) gave customers greater license to purchase vice categories (e.g. ice-cream).

A disadvantage of those studies using RFID technologies such as Path-Tracker is that the tags are attached to carts and baskets. Paths taken by shoppers walking without carriers are not recorded. In the supermarkets where many of these studies were performed, very few customers do not use a cart or basket, but in many other retail environments—such as the Best Buy store studied in this thesis—customers peruse the store without accoutrement. Alternative methods for tracking customer movement are required. We gathered customer trajectories using video collected from overhead cameras for the case study retail store. We will return to video based tracking shortly, after a discussion of more general pedestrian movement models.

## 2.3 General Models of Pedestrian Movement

Another thread of analysis which aims to make predictions of pedestrian behavior from the shape and configuration of the environment is *Space Syntax*. This subfield at the intersection of architecture and sociology was inaugurated by Bill Hillier in (Hillier et al., 1976) and expanded in *The Social Logic of Space* (Hillier and Hanson, 1984) (an accessible overview of the ideas of space syntax analysis can be found in Bafna, 2003). The theory posits that the interconnectivity of spaces afforded by movement or visibility both reflects the functions of the space[5] and influences the behavior of its users. The analysis transforms the plan of a building or urban environment into an abstract graph which can be analyzed computationally, deriving measures such as *depth* (the mean distance travelled by shortest route to each of the other nodes in the graph), *choice* (a measure of the number of alternate routes from one space to another), and *integration* (the reciprocal of the mean number of nodes traversed to travel from a node to each other node.) (March and Steadman, 1971; Hillier et al., 1987). These measures correlate with the functions of the space—a hallway will have high *integration*, a bathroom low *choice*. The graph structure describing a building or cityscape is formed by the intersection of *axial lines*, long vistas which connect distinct physical spaces. Hillier and Hanson's original definition of the axial line, and the axial map are procedural (Hillier and Hanson, 1984). Writing about their use in the analysis of settlements:

> Next make an axial map of the settlement by first finding the longest straight line that can be drawn in the $y$ and drawing it on an overlaid tracing paper, then the second longest, and so on until all convex spaces are crossed and all axial lines can be linked to other axial lines without repetition are so linked. (p. 99)

The definition was later formalized in Carvalho and Batty (2003) and Turner et al. (2005). An example axial map of a building is shown in Figure 2.2.

In the past decade, viewshed analysis has become a prominent tangent to space syntax. Rather than considering a macroscopic unit such as a maximal volume—roughly speaking, a room—or axial line of visibility, visibility analysis derives quantitative measures at every point in a space from its *isovist* or *viewshed*, the shape defined by the set of all points visible from

[5] *Function* here refers to the occupant's use of the space; in the nomenclature of architecture, the program. For example, a hallway's function as a passage from one place to another, or a doctor's waiting room as a buffer and gateway.

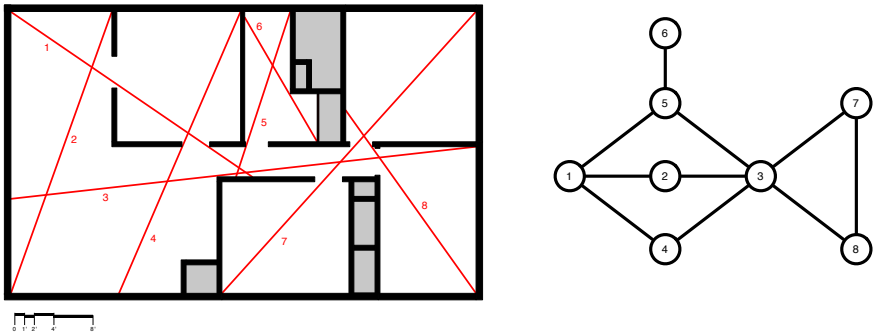


**Figure 2.2:** The axial map of a single-family residence, and the corresponding axial graph. Nodes in the graph represent axial lines. There exists an edge between every two axial lines (nodes) which intersect. Measures such as *depth*, *choice*, and *integration* are calculated using this graph.

a vantage point (See Figure 2.3). These measures were shown to correlate to qualitative experience (Wiener and Franz, 2004) and also mental representations (Meilinger et al., 2009).

Whereas the space-syntactic and viewshed analysis place emphasis on the effect of a space's geometry on pedestrians within, another approach takes an egocentric perspective. These models of human movement can be categorized as flow, cellular, and agent-based. Flow models (for example Henderson, 1971) make an analogy between the discrete individual actions of people and continuous physical phenomena like the flow of a fluid——individuals are modeled essentially as particles of a gas or incompressible fluid. Cellular models discretize a physical space into a collection of connected cells whose state changes as a function of local interactions with neighboring cells (Burstedde et al., 2001; Kirchner and Schadschneider, 2002; Kaneda and Suzuki, 2005). In agent-based models, individual people (or cars, ants, etc.) are modeled as independent entities, each capable of sensing its surrounding environment, and choosing actions as a function of both internal and external state. Agent-based models operate in both discrete (e.g. Gipps and Marksjö, 1985) and continuous environments (e.g. Hoogendoorn, 2003)—in the former, they resemble cellular automata models.

Of the three classes of pedestrian models—flow, cellular and agent-based—agent-based models have been most effective matching the microscopic movements of people in buildings and cities. Social-force and vision-based models of motion are among the more successful. In the former, other pedestrians, obstacles and objects in the environment apply "forces" on the modeled

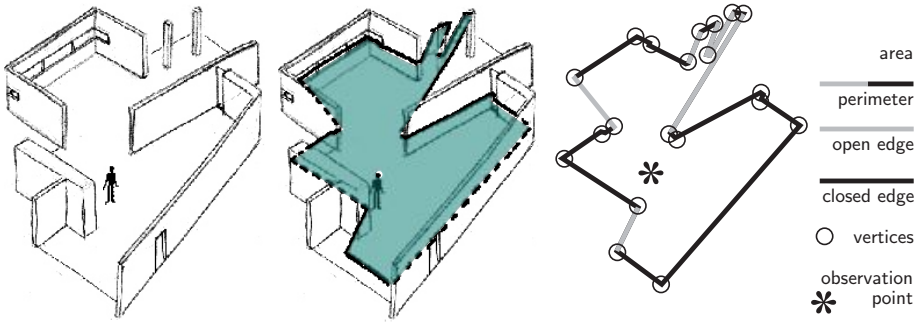


**Figure 2.3:** In this illustration (from Wiener and Franz, 2004), an isovist (viewshed) is drawn for a hypothetical environment. The isovist is the polygon enclosing the visible areas from a particular vista.

person (Helbing, 1991; Helbing and Molnár, 1995). Social force models have been used to predict the dynamics of escape panics (Helbing et al., 2000), the spontaneous forming of lanes of motion (Helbing et al., 2001), trail systems (Helbing et al., 1997), and to recognize anomalous crowd behavior (Mehran et al., 2009). In vision-based agent models such as Turner and Penn's, pedestrians choose their next step based on the places currently visible (Turner and Penn, 2002). Recently, a model based on the affordances of visibility was shown to match pedestrian movement at a finer level of detail (Moussaïd et al., 2011). A very thorough review of pedestrian models can be found in Winnie Daamen's PhD thesis (Daamen, 2004). She develops a detailed agent-based pedestrian model for public transit facilities named SimPed, which models walking, route-choice, alighting and activity. The model is validated and calibrated using pedestrian traffic data gathered at a public transit station in Delft.

Video has often been used to validate models of pedestrian movement. Experiments are often performed in highly controlled environments with pedestrians sometimes physically tagged to facilitate their tracking during analysis (See, for example Hoogendoorn et al., 2003; Hoogendoorn and Daamen, 2005; Moussaïd et al., 2009). Support for pedestrian models also comes from video captured in unconstrained settings such as from city streets (Willis et al., 2004) and public transit hubs (Berrow et al., 2005). Bauer and Kitazawa also validated and calibrated a social-force pedestrian model in a controlled setting, but used laser scanning instead of using video to capture data (Bauer and Kitazawa, 2010).

## 2.4 Tracking People

Validating and calibrating pedestrian models by tracking people evokes the much broader field of person tracking. The most basic way to track a person's movements in a building is by following them, clipboard in hand, recording their paths manually. One of the early and oft cited studies which used this simple yet expensive technique is an analysis of the Tate Gallery, Millbank, performed by the Bartlett School of Graduate studies (Hillier et al., 1996). This study manually traced ninety-three gallery patrons for ten minutes, gathering a dataset that has subsequently been used in several space-syntax publications aiming to connect the paths to the social logic of the building (Turner et al., 2001; Wiener and Franz, 2004) as well as validate the behavior of vision-guided pedestrian simulations (Turner and Penn, 2002).

Radio-based technologies, like the RFID PathTracker system described earlier and the WiFi used in Uotila and Skogster (2007), have been used to actively track human movement in buildings. Mobile phones offer an alternative, passive tracking approach (Bourimi et al., 2011). Rather than tracking an individual person using tags as in the above examples, Ivanov et al. used a network of closely-spaced motion-sensors to reconstitute coarse-grained paths in a building (Ivanov et al., 2007). Browarek also used passive thermal sensors to track individuals in a room; her system is able to localize individuals at much higher resolution (Browarek, 2010).

Tracking objects—often people—is one of the earliest and best studied subproblems in computer vision. An excellent overview of recent progress in tracking techniques can be found in (Yilmaz et al., 2006); an earlier survey of tracking, specific to human activity, can be found in (Moeslund and Granum, 2001). The Mimic system described in this thesis uses a video-based tracker developed by George Shaw (Shaw, 2011). An overview of its operation can be found in Appendix A. One of the greatest obstacles to accurate tracking in Shaw's system is the frequent occlusions caused by people walking or standing behind fixed objects such as furniture, or other tracked people. Techniques for coping with occlusions is an active subfield of computer vision research. An early review of approaches to the occlusion problem can be found in Gabriel et al. (2003). Recent examples of tracking under occlusion include color appearance-based models such as Yang et al. (2005) and Senior et al. (2006).

Mimic relies on the trajectories of customers in stores. The techniques discussed above can be used to gather such data. Mimic uses trajectories to classify high-level behaviors; in this sense, the system is one which performs video event understanding.

## 2.5 Video Event Understanding

Computational models that deduce human behavior from video, often called video event understanding, is a lively subfield in contemporary computer vision. Typically, these computational models cover domain-specific behavior classification, for example walking/running/waving, sign-language, making-breakfast/having-a-snack, etc. Lavee et al. broadly categorize these event understanding strategies into pattern recognition methods, state models, and semantic models (Lavee et al. 2009, see also Turaga et al., 2008 for a very complete review). Among the pattern recognition models, one finds discriminative methods such as nearest-neighbor (Zelnik-Manor and Irani, 2006), neural networks (Vassilakis et al., 2002) and support vector machines (Pittore et al., 1999). State models such as finite-state machines (Hong et al., 2000), Bayesian networks, Hidden Markov Models and conditional random fields (Sminchisescu et al., 2006), try to capture the temporal dynamics of a hidden state (often, the behavior in question). Finally, semantic methods collapse the often huge state-space of state models by constraining possibilities to those acceptable by a externally imposed semantic model (Borzin et al., 2007; Kitani et al., 2007; Siskind, 2000). Examples of these semantic models include petri-networks, grammars, logic and constraint satisfaction.

Perhaps closest in spirit and mechanism to the work of this thesis is Fleischman's method for classifying behavior from video (Fleischman et al., 2006). In Fleischman's system, overhead video in a kitchen is automatically classified into macroscopic behaviors such as "eating breakfast" or "making coffee" by composing motion in several hand-labeled regions of the video frame (analogous to the functional-locations described in the next chapter) into a hierarchy of temporal relationships. Hierarchies representative of high-level behaviors are then learned by a tree-kernel support vector machine. As in Mimic, classifications are made for patterns of activity which may involve several people.

## 2.6 Commercially Available Retail Video Analytic Systems

Retailers have long employed video for loss prevention and more recently begun using their deployed infrastructure to better understand the operation of their stores. Inexpensive computing power and digital storage, and the easy availability of cutting-edge computer vision through open source libraries

such as OpenCV[6] has made retail video analytics a burgeoning commercial space. Simple doorway people-counters using infrared beams have given way to computer-vision based counters that use visible light and thermal infrared.[7] Video is now used to measure dwell time and occupancy, manage queues, and gather customer demographics. Verint,[8] a representative video analytics company, uses video to calculate conversion-rates on a per-product basis, find trends in customer visits with hourly and weekly granularity, and measure the effects of advertisements and promotions. Other companies involved in retail video analytics include Lighthaus Logic,[9] Scopix Solutions,[10] GfK,[11] Experian Footfall,[12] and Tyco International.[13]

## 2.7 Putting It All Together

This chapter has examined several of the research fields this thesis intersects. We are interested in the behavior of customers in stores—present in both sociology and marketing research. Unlike the human observation methodologies often employed in these fields, the Mimic model of this thesis uses customer trajectories automatically gathered from video footage. The following chapter will introduce *functional locations* as a mechanism to simplify paths, a technique which echoes some previous modeling of path data.

From computer vision to ethnographic investigation, the physical scale of study falls along a broad spectrum. Vision-based classifiers of race and gender focus on human features measured in centimeters; other models of mobility which use similar techniques as Mimic operate at an urban scale. We focus on a middle-ground—the scale of a small to mid-sized store—where measures are made on the order of feet (Figure 2.4).

Much of contemporary research most closely related to this thesis focuses on supermarkets. Grocery stores are a good research target: customers buy a wide variety of goods, staples and impulse purchases, and the larger size of the

[6] `http://opencv.willowgarage.com/wiki/`
[7] See: Shoppertrak (`http://www.shoppertrak.com`)
Countwise (`http://www.countwise.com`),
SenSource (`http://www.sensourceinc.com/peoplecounters.htm`),
Honeywell (`http://www.honeywellvideo.com/products/ias/va/160978.html`),
Sensormatic (`http://www.sensormatic.com/Products/StoreBusinessIntelligence2`).
[8] `http://verint.com/corporate/`
[9] `http://www.lighthauslogic.com`
[10] `http://www.scopixsolutions.com`
[11] `http://www.gfkamerica.com/sectors/consumer/shopper_insights/`
[12] `http://www.footfall.com/`
[13] `http://www.americandynamics.net`

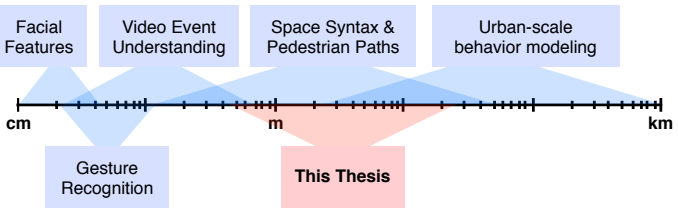


**Figure 2.4:** The scale of study of the disciples touched in this thesis, and where Mimic fits. At the most microscopic scale are ethnographic studies and some computer vision research, which examines body language and facial expression. At the other end of the scale are models which use path data gathered on a city-sized scale. This thesis sits in the middle, useful for examining behavior at the scale of buildings.

stores gives credence to the hypothesis that purchase decisions are intimately linked to the configuration of the space. Moreover, nearly 20% of US retail sales occur in food and beverage stores (US Census Bureau, 2011c). Though a sizable fraction, a large portion of the retail world remains under-explored. As we shall see in Chapter 4, Mimic can be used to model customer behavior in a much more challenging setting than supermarkets, a smaller store hosting higher-priced products much less susceptible to impulsive purchases.

Many techniques discussed in this chapter have potential to be incorporated as components of Mimic with the likely consequence of improving the predictive power of the model. This is especially true for the computer vision techniques. Robust cross-camera tracking of individuals in the store would enable predictions local to an individual, rather than global to the state of the store as a whole. The development of Mimic was guided by observations and conclusions made by sociologists and ethnographers. Completing the circle, Chapter 6 discusses the opportunities for ethnographers enabled by Mimic.

Chapter 3

# MIMIC: Prediction Using Functional Locations

The experienced store manager looking out onto her store has an intuition of the buzz of activity. She knows if customers are focused and intent on finding the best deal, if the slow wander of a man and his baby carriage is just biding time, if the children nagging and running from one display case to another will be appeased by a parent's purchase of a distracting toy. We wish to embed some simpler form of this intuition in a computational model. Like the manager looking onto the floor, our model will take in the aggregated activity of the whole store and predict an aspect of the patrons' behavior—the likelihood of a purchase. This computational model is named MIMIC.

The model takes as input the aggregated activity of the whole store represented by the trajectories of the store's patrons. The data is then decimated and dimensionally reduced by discarding those parts not functionally relevant to a purchase. To classify a period of activity as preceding purchase, the episode is evaluated by a pair of generative models and a label chosen from their results. Hidden Markov Models (HMMs) express the dynamic pattern of activity, using one of several state-models to represent static activity distributions.

In this chapter, I introduce *functional locations* as a technique for creating a homogeneous and dimensionally reduced feature set that captures the patterns of activity in a building. Next, I describe the static and dynamic models of activity that operate on this reduced feature set. Finally, I detail the training of the model and the way it is used to classify purchases.

## 3.1 Functional Locations

What is common between a library, train station, store and hospital ward? Each of these spaces is designed to fulfill a small collection of its occupants' goals. Library patrons may want to *find-a-book*, *check-out* or *attend-a-reading*. Travelers in a station want to *get-to-their-platform*, *buy-a-newspaper*, *purchase-tickets*, or simply, *wait*. Shoppers want to *browse* or *purchase*. Nurses need to *do-rounds* and *attend-to-calls*. These activities are far from exhaustive, of course, but illustrative of the goal-directed nature of the space. Contrast these spaces with homes, fairgrounds or convention centers which serve a large collections of uses.

Naturally emergent from these goal-directed spaces are what I call *functional locations*, or *flocs*: small areas within the space directly or indirectly relevant to the tasks/goals in the space. The ticket counter in the train station is a functional location because it is relevant to the task of riding a train. A bench is a floc, relevant to the task of repose. A ribbon-barrier is relevant to the task of queueing. The core hypothesis of this thesis—that patterns of activity in the whole of a store are predictive of an intent-to-purchase—is a specific instantiation of the intuitive observation that patterns of human activity within a collection of flocs is predictive of the high-level behavior of the occupants. For example, there may be a pattern of activity in the hospital ward, a signature, which is emblematic of a staff's preparation for disaster-response, even if the individual activities in individual functional locations are common across different macroscopic behaviors.

A human investigator or designer should be able to quickly identify the functional locations in a space from common or domain-specific knowledge. Flocs in this thesis were manually annotated; manually-chosen points-of-interest have also been used to reduce trajectories to computationally manageable form in several earlier works (e.g. Fleischman et al., 2006; Hui et al., 2009b). Flocs can also be determined in a data-driven manner by clustering activity observed in video, the tracked location of pedestrians, or other measures. In his masters thesis, Matthew Miller clustered regions in video captured in a single-family home based on activity derived from motion within the frame (Miller, 2011). The automatically discovered regions roughly correspond to what a human annotator might label as functional locations. For example, in the kitchen, clusters around the sink, refrigerator, and cabinets were automatically discovered. The strength of using a data-driven approach to place functional locations is that the floc contours conform to actual use, and the unforeseen uses of a location can be discriminated and incorporated. However, automatically discovered regions such as those

derived by Miller may be harder to integrate from multiple video sources, and are less malleable to experimental variation, one of the high-level outcomes of this thesis (described in greater detail in Chapter 6).

How can we define a functional location? A floc is not a single point in space, but rather a volume encompassing a *zone of interaction.* The bench floc in our train station includes both the sitting portion, as well as the area immediately surrounding it, where a family might congregate with their children. The boundaries of a functional location are elastic—when the family departs the bench, its area shrinks so the lone traveler, stopping to check her Blackberry, is no longer encompassed. The extent of a floc is fuzzy in the same way that the requirement for functional relevance is vaguely defined. Take for example, the signage showing train schedules and track numbers in a train station. The sign clearly has a functional relevance to the task of boarding a train, but where are its extents? Would some arbitrarily chosen units of radius best describe the volume? Is its radius whatever distance from which the sign can be read? Or should the floc be defined based on observed use, capturing the statistics of where pedestrians peer up to determine the status of their train?

To make concrete this notion of functional locations, I propose the following four criteria for a floc. Flocs...

1. enable a person to accomplish a goal or subpart of a task tied to some function of the room or building.

2. are three-dimensional volumes which a person can occupy.

3. are anchored to a fixture in the space.[1]

4. are sized large enough to capture the activity of humans within, and small enough to enclose a group focused on a single task.

Of these characteristics, all but the first may be easily encoded in a computer. Deciding whether a place affords something task-relevant is a task for which a human researcher/designer is best suited.

Human behavior in buildings is enormously varied; at a high level, a whole vocabulary of verbs can be used to describe our actions. At a low level, behavior is dynamic and continuous; we are almost continuously in motion, fidgeting, walking, sprinting. Flocs can help code observed behavior as it relates to the physical space. Flocs do this, as we will see later in this

[1] That is not to say that flocs are always stationary. Furniture (a chair, couch) is often moveable. The floc, anchored to the furniture, moves with it.

chapter, in a way easily integrated into computational models. They help simplify the complex, varied, continuous and dynamic behavior of people into discrete data more easily integrated into a host of downstream models.

## 3.2 The Activity Vector

Functional locations are a mechanism for reducing the dimensionality and complexity of features derived from video into a form useable by machine learning algorithms. Here, the high dimensional and complex derived features are the trajectories of customers and employees within the store. A track which describes such a trajectory, $\tau$, is a collection of timestamped observations of the state of a person within the store.[2] This state includes the position as well as (optionally) secondary characteristics such as size, color distribution, velocity, etc.

We can encode the distribution of activity—the gestalt—of the store over a duration in a single high-dimensional feature vector $\vartheta$ with $n$ dimensions, one dimension for each floc in the store. To calculate $\vartheta$ for a short duration from a collection of tracks, first find all tracks which exist during the target duration. For each of these, look at which flocs the tracks pass through and add the time spent in each of those flocs during the interval to the associated dimension in $\vartheta$. More formally, for functional locations $\{\upsilon_1, \ldots, \upsilon_n\}$, tracks $\{\tau_1, \ldots, \tau_m\}$, and time period $[t_1, t_2]$, the value $\vartheta^i$—the value in dimension $i$ is:

$$\vartheta^i = \sum_{j=1}^{m} \{\text{duration } \tau_j \text{ spent in } \upsilon_i \text{ between } t_1 \text{ and } t_2\}$$

This vector captures the distribution of *occupancy* in the store. The simple occupancy measure serves as a proxy for a person's engagement with the function of the location. With more refined source data than gross-level trajectories, one could imagine using other relevant measures of activity within a floc. For example: interaction with key physical elements or other people within the floc, the physical gestures exhibited, or affect as derived from facial features.

The vector $\vartheta$ codes the *static* distribution of activity in the store—the state of activity during brief moment in time. How this distribution evolves over time, the store's *dynamic* distribution, is coded by a sequence of $\vartheta$ for consecutive periods of time. For clarity and simplicity of notation, I use a fixed timebase ($\Delta t$) to divide time, so $\vartheta_t$ represents the feature vector

[2] A glossary to the variable terminology used in this thesis can be found in Table 3.2.

encoding activity between time $t$ and $t + \Delta t$. The interval $\Delta t$ is short—on the order of several seconds; on a timescale that captures a unit of human action such as reaching to pick up an item, or manipulating an object.[3]

The activity vectors may suffer from edge effects resulting from the choice of timebase. Additionally, for short timebases and dispersed flocs, the activity vectors many have many zeros as pedestrians walk between flocs. Many zeros often causes the downstream model to converge prematurely to a non-optimal state. One way to minimize the consequence of these edge effects and zeros is by smoothing, averaging $\vartheta_t$ with its temporal neighbors. Smoothing also allows highly granular data to be further reduced by subsampling the smoothed sequence.

Let's step back and define what is meant by the "pattern of activity" that is coded by these feature vectors. Each activity vector describes a snapshot of *where people are* in the store with regards to the places that matter in decision-making. A sequence of these snapshots shows how this distribution evolves but discards a potentially revealing characteristic: who went where. Extracting an accurate trace of a person's movements from video is a significant challenge; in the case of multi-camera settings such as that in the Mimic case study, several common failure-modes of tracking and cross-camera handoffs make accurate complete trajectories nearly impossible. Moreover, some of these failure modes, such as a single track which incorrectly jumps from one person to another, could be significantly damaging to those models incorporating an individual's passage through multiple functional locations. The activity vector $\vartheta$ is robust to these types of errors yet still represents an important facet of overall behavior within the store. The vector codes not just the quantity of activity, but also where people are *relative to each other*. A sequence of activity vectors is a compact and interpretable representation which filters out likely irrelevant motion and enables a wide range of downstream machine learning techniques.

## 3.3 Models of the Static Distribution

A generative model of the static distribution calculates the likelihood of the distribution $\vartheta$. Below are three proposals for static activity distributions. Simplest is a binomial model which looks for merely the *presence* of activity, and makes a key independence assumption. The multinomial model also takes into account the magnitude across each dimension. Finally, a mixture-of-Gaussians model considers the distribution of activity as a point in a

[3] In the case study described in the next chapter, a five-second interval was used.

very high dimensional space (whose dimensionality is the number of different flocs). Each of these models has a tunable free parameter, and a collection of parameters learned from the data.

**Binomial:** This model ignores the total activity in any dimension and only considers the presence of activity. Each dimension in the feature vector is modeled as independent. The model's parameters are a threshold $\alpha$, and a set of probabilities, one for each dimension of the activity vector $\vartheta$:

$$\theta_{\text{binom}} = \{\alpha, p_1, \ldots, p_n\}$$

The value $p_i$ represents the probability that the activity in dimension $i$ is greater than the threshold $\alpha$ (typically zero). So the likelihood of an activity distribution $\vartheta$ is:

$$P(\vartheta|\theta) = \prod_n \begin{cases} p_i & \text{if } \vartheta_i > \alpha \\ 1 - p_i & \text{otherwise} \end{cases}$$

In other words, the activity vector $\vartheta$ is transformed into a binary feature vector $\vartheta'$ by passing each dimension through an indicator function

$$\mathbf{I}(x) = \begin{cases} 1 & \text{if } x > \alpha \\ 0 & \text{otherwise} \end{cases}.$$

then

$$P(\vartheta'|\theta) = \prod_n \left\{ p_i \vartheta'_i + (1 - p_i)(1 - \vartheta'_i) \right\}$$

The observation space here is the binary feature vector $\vartheta'$.

Training the binomial model is trivial: given training data of many distribution examples, the probability for each dimension is estimated using counts. A Laplace correction is added to prevent zero probabilities and smooth the model.

In summary, the binomial activity distribution model has one free parameter: the threshold $\alpha$ which the activity within a floc must exceed to be counted as a positive activation.

**Multinomial:** The multinomial is a natural extension of the binomial model which incorporates the magnitude of activity. Multinomials, like binomials, are not suited for modeling continuous values. The input activity vector, which may contain real values (e.g. fractions of seconds), is first discretized

into counts of small time-units each of duration $\delta$: discrete quantums of activity.

As before, the real-valued feature vector $\vartheta$ is first transformed, this time into a integer-valued feature vector $\vartheta$ (which makes up the observation space) by passing each dimension through the discretization function

$$D(x) = \left\lfloor \frac{x}{\delta} \right\rfloor .$$

As with the binomial, the multinomial is parameterized with a set of probabilities:

$$\theta_{\text{mulinom}} = \{\delta, p_{\text{zero}}, p_1, \ldots, p_n\} \quad \text{such that:} \quad \sum_1^n p_i = 1, \quad 1 > p_{\text{zero}} > 0$$

Multinomials are only well defined when the trial count (here, the total number of activity quantums, $\sum_n \vartheta'_i$) is greater than zero; Since it is possible that $\vartheta' = 0$, this model is two-staged. If $\vartheta' = 0$, the likelihood is fixed to $p_{\text{zero}}$, a parameter of the model. Otherwise, the likelihood of an activity vector is calculated as:

$$P(\vartheta'|\theta) = (1 - p_{\text{zero}}) \cdot c! \prod_{i=1}^{n} \frac{{p_i}^{\vartheta'_i}}{\vartheta'_i!} \qquad \text{where} \qquad c = \sum_{i=1}^{n} \vartheta'_i$$

Training the multinomial model is again simple. From training data, $p_{\text{zero}}$ and $p_1, \ldots, p_n$ are estimated by counting. As with the binomial, a Laplace correction is added to smooth the distribution and prevent zeros in the distribution probabilities.

**Mixture of Gaussians (GMM):** This model estimates the distribution of activity as a mixture of several weighted Gaussian Normal functions, each with different mean and covariance. The real-valued activity distribution $\vartheta$—the observation space of this model—is a sample drawn from this high-dimensional distribution.

The free parameter to this model is $k$, the number of gaussian mixtures. A parameterization of this model is defined as:

$$\theta_{\text{gmm}} = \{w_1, \ldots, w_k, \mu_1, \ldots, \mu_k, \Sigma_1, \ldots, \Sigma_k\} \quad \text{such that:} \quad \sum w_i = 1$$

| Model | Free Parameters | Learned Parameters |
|---|---|---|
| Binomial | $\alpha$ | $p_1, \ldots, p_n$ |
| Multinomial | $\delta$ | $p_{\text{zero}}, p_1, \ldots, p_n$ |
| Gaussian Mixtures | $k$ | $w_1, \ldots, w_k, \mu_1, \ldots, \mu_k, \Sigma_1, \ldots, \Sigma_k$ |

**Table 3.1:** Free and learned parameters of the three static distribution models.

The likelihood of an activity vector $\vartheta$ is calculated as:

$$P(\vartheta|\theta) = \sum_{i=1}^{k} w_i \cdot N(\vartheta; \mu_i, \Sigma_i)$$

where $N(\vartheta; \mu, \Sigma)$ is the standard multivariate normal distribution of dimension $n$ with mean $\mu$, and covariance $\Sigma$:

$$N(\vartheta; \mu, \Sigma) = (2\pi)^{-\frac{n}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\vartheta - \mu)'\Sigma^{-1}(\vartheta - \mu)\right)$$

The parameters $w, \mu$ and $\Sigma$ are learned using expectation maximization (Redner and Walker, 1984). A minimum variance is enforced during training to prevent over-fitting due to individual training examples forcing mixture variances to zero.

A mixture model is more difficult to interpret because of the high-dimensional space it inhabits. An example may help an intuitive understanding. Imagine a room in a museum, with flocs beside the works of art in the room. Tour groups routinely visit several of the works in the gallery, with several people crowding into the art's respective flocs as they move about. A model trained on data derived from these tour-group visits would ideally align individual Gaussians in the mixture model with means centered on each of the flocs visited by the tour-groups, and variance chosen to capture the gross activity.

Table 3.1 summarizes the free and derived parameters to each of the static models.

## 3.4 Capturing the Dynamic Distribution

Mimic models the temporal evolution of activity distributions using a standard Hidden Markov Model (HMM) (Rabiner, 1989), calculating the likelihood of a sequence of activity pattern, $\{\vartheta_1, \vartheta_2, \ldots\}$. The sequential series of

activity vectors, $\{\vartheta_1, \ldots, \vartheta_m\}$, I call an *episode* and denote $V$. In an HMM, the system is assumed to be in one of several hidden states, $\{s_1, \ldots, s_z\}$, from which samples may be drawn. At each time step, a sample is drawn from the current state, and the HMM transitions to the next state with a probability governed by the weighted, directed edges of a graph connecting the states. The free parameters to the HMM are the number of states, the topology of the state graph, and the generative model representing a state. The transition probabilities and the initial state probabilities are learned parameters.

The functioning of the model is best understood in its generating capacity—drawing a sample from its distribution. Here, a sample is a time series of floc distributions, $V$. At each time step $(t)$, the system is assumed to be in a single state, $s_t$. The first state is chosen based on the state priors. A sample is drawn from state model $s_t$ and then a transition to the next state $(s_{t+1})$ is made based on the transition probabilities encoded in the arcs of the graph. The likelihood of a generated sequence equals the product of the probabilities of state-transitions used to generate the sample and the likelihoods of the samples as generated by the state sequence.

$$\begin{aligned} P(\vartheta_1, \ldots, \vartheta_m | s_1, \ldots, s_m, \theta_{\text{HMM}}) = P(s_1 | \theta_{\text{HMM}}) \\ \cdot \prod_{i=1}^{m-1} \{P(\vartheta_i | s_i) \cdot P(s_{i+1} | s_i; \theta_{\text{HMM}})\} \\ \cdot P(\vartheta_m | s_m) \end{aligned}$$

Inference using the model—finding the most likely sequence of hidden states given a sequence of sampled activity vectors—is done using the Viterbi algorithm (Rabiner, 1989). The likelihood of the sequence can then be calculated as above.

Given a set of training data, the parameters to the HMM are learned using expectation maximization (EM). The prior probabilities of the states are first set to uniform, and each state is initialized with training examples drawn without replacement from the training data. Though the EM algorithm increases the likelihood of the training data with every iteration, it is prone to long periods of nearly flat performance improvement—getting stuck in local minima—often leading to the premature termination of the algorithm. Constraining the topology of the HMM's state transition graph can improve EM's performance if the topology better matches the underlying true distribution. In the experiments that follow in Chapter 5, I used several graph structures to constrain the dynamic structures captured by the HMM

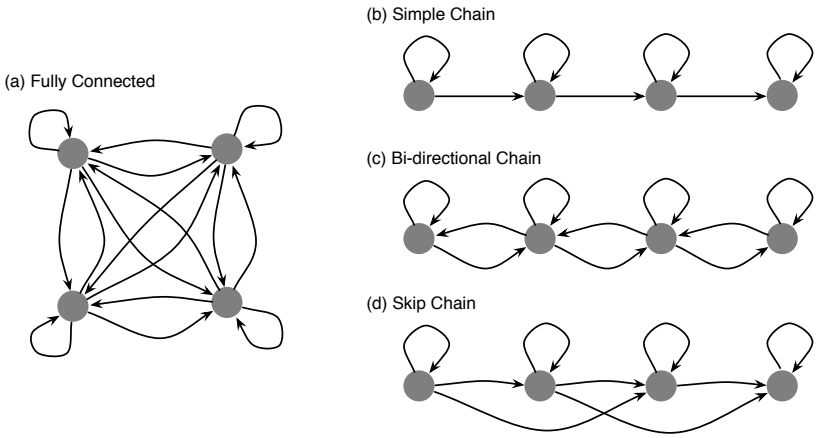


**Figure 3.1:** The HMM graph topologies tested.

in an attempt to encourage the model to converge to a better representation of the data (See Figure 3.1). These graph structures are equivalent to constraining a fully-connected model by forcing some state transition probabilities to be zero. The intuition behind these topologies makes analogy to the organization of an individual's behaviors in a store. We can imagine that the hidden states correspond to high level behaviors—browsing, questioning, searching, waiting, purchasing, etc.—for which there are temporal constraints which can be encoded in graph topology. Whether an analogy may be made between an individual and the group's behavior will be borne out in the model's performance. The topologies tested were:

**Fully connected:** This graph structure connects every state to every other state, including self-transitions. It is the most general topology.

**Simple Chain:** Each state $s_i$ has two outgoing transitions, one to itself, and one to the next state, $s_{i+1}$. This structure is well suited to capture linear steps in the activity pattern, for example, a "browsing" state leading to a "purchasing" state.

**Bi-directional chain:** Like the simple chain with each state (except the first) having an addition outgoing transition to the previous state.

**Skip chain:** Also similar to the simple chain, but each state has three outgoing transitions: to itself, to the next state, the one following that, $s_{i+2}$.

| Variable | Use/Meaning |
|---|---|
| $t$ | Time. |
| $\tau$ | A track: the trajectory of a person in the store. |
| $\upsilon$ | A functional location (floc). |
| $\vartheta$ | A floc activity vector. The distribution of activity among $n$ flocs over a duration. |
| $V$ | A sequential series of activity vectors. An episode. |
| $k$ | The number of mixtures in the Gaussian mixture model. |
| $w$ | The weight associated with a single Gaussian mixture. |
| $m$ | The number of activity vectors ($\vartheta$) in a sequence. |
| $s$ | A state in the HMM. |
| $z$ | The number of states in the HMM model. |
| $\psi$ | The duration of an episode used for training data. |

**Table 3.2:** Glossary to the nomenclature and variables of the Mimic model.

The increased complexity of this graph structure can support either-or type patterns. For example, a "browsing" state (say, $s_1$) can transition to either a "purchasing" state (say, $s_2$) or an "inquiry" state, ($s_3$).

To summarize the dynamic model, we have an HMM with three free parameters: the number of states, the type of state model (binomial, multinomial, or GMM), and the structure of the state transition graph. The learned parameters of the model are the state priors, the transition probabilities and the parameters of the chosen state model.

## 3.5 Classification Using Mimic

We wish to use the dynamic, generative model of activity patterns to build a discriminative classifier which labels episodes as either indicative of a transaction taking place, or not.[4] To classify using these models, we train two HMMs, one positive (preceding transaction) and one negative (not associated with a transaction). An activity pattern is classified by calculating the likelihood of the data given each of the two models, then thresholding the ratio of the two model likelihoods. Above threshold, the example is labeled as positive; below threshold, the example is labeled negative. The value

[4] The rest of this discussion focuses on binary classifiers, though the generalization to multi-class models is straightforward.

of this threshold sets the burden of evidence for an episode to be labeled as either positive or negative. For simplicity, the number of states in both positive and negative HMMs are forced to be the same.

Formally, the parameters to the classifier are $\{\theta_{\text{pos}}, \theta_{\text{neg}}, \gamma\}$, where $\theta_{\text{pos}}$ and $\theta_{\text{neg}}$ are full parameterizations of the positive and negative HMM models, and $\gamma$ is the likelihood-ratio threshold. An episode $V$ is classified as

$$\text{label}(V) = \begin{cases} \text{“positive”} & \text{if} \quad \frac{P(V|\theta_{\text{pos}})}{P(V|\theta_{\text{neg}})} > \gamma \\ \text{“negative”} & \text{otherwise} \end{cases}$$

This formulation is equivalent to a simple likelihood comparison with a probabilistic prior. Assuming the prior of a positive class is $\rho$, then a positive label is given to an episode if

$$\rho \cdot P(V|\theta_{\text{pos}}) \quad > \quad (1-\rho) \cdot P(V|\theta_{\text{neg}})$$

rewritten:

$$\frac{P(V|\theta_{\text{pos}})}{P(V|\theta_{\text{neg}})} \quad > \quad \frac{1-\rho}{\rho}$$

which shows the equivalence of the threshold $\gamma$ and the prior ratio $\frac{1-\rho}{\rho}$.

In summary, classification using the dynamic model uses one free parameter, the likelihood ratio threshold $\gamma$, and a pair of models of dynamic activity patterns with their respective parameterizations, $\theta_{\text{pos}}$ and $\theta_{\text{neg}}$.

## 3.6 Training the Model

The positive and negative HMMs used for classification are each trained with a different set of episodes. Which episodes should be included in each set? Supposing we have the transaction record from a store—both the contents of a transaction and the timestamp of occurrence—and also the recorded activity patterns from an entire day. From this data we do not know the exact amount of time a customer was in the store before engaging in a transaction. Moreover, this duration is different for each transaction; some may be quite brief (buying AAA batteries from a kiosk near the register), while others many minutes long (activating a cell phone with a new provider). Since this information is not available in the training data, I chose fixed length windows preceding a transaction to serve as positive examples of transaction-type activity (See Figure 3.2). The negative space of this set—

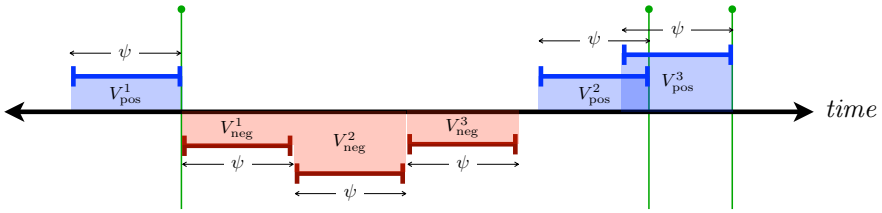


**Figure 3.2:** Training data derived from timestamped transactions and a long time-series of in-store activity. In this schematic illustration, time lies on the $x$ axis; vertical marks indicate timestamps of transactions. Here, the three positive episodes are marked above the time-line. Negative examples are marked below. Note that the two training examples $V^2_{\text{pos}}$ and $V^3_{\text{pos}}$ overlap, resulting in some duplicated data.

all data not encompassed within these positive examples was broken into examples of non-transaction-type activity. The duration of the episode window, $\psi$, is an additional parameter to the classifier. A consequence of this simplification is that positive example may include a fraction of data that should be labeled negative, and vise versa. With the ability to trace a transaction to the moment when the transacting customer entered the store, both positive and negative examples could be scrubbed of this contamination. Since oftentimes, several transactions may take place within a window there is some replication of the training data. Chapter 5 discusses the method used to prevent cross-contamination of training and testing datasets that can result from the episode overlap.

The duration $\psi$ is a coarse means to derive training data; better would be to use the exact duration of a customer's stay in a store prior to their participation in a transaction. The fixed value for $\psi$ has the additional consequence of changing the ratio of positive and negative example counts. A small $\psi$ leaves more of the day's data available to be divided and used as negative training examples; equivalently, a large $\psi$ leave less "negative space" available to be divided for use as negative examples.

## 3.7 Summary

Mimic classifies episodes of in-store activity by first extracting features from functionally relevant locations, then calculating the likelihood of those features in each of the label scenarios. Likelihoods are calculated using a traditional Hidden Markov Model, with one of several generative state models

which capture the static distribution of activity.

The concept and use of functional locations operationalizes the hypothesis of this thesis: that patterns of activity within a store are characteristic of customer intent-to-purchase. The next chapter presents a case-study experiment where the Mimic model will be put to test. Chapter 5 evaluates the performance of the model on real data captured at this operating store.

Chapter 4

# A Case Study

The Mall of America, located near Minneapolis in Bloomington, Minnesota, is the largest mall in the United States, with 4.2 million square feet of gross area.[1] It is square in shape, with each side roughly the size and configuration of a typical American suburban mall: three stories in height, with a long central hall and stores at either side. The center of the square is a glass enclosed space with a small amusement park, complete with several roller coasters. The mall attracts forty million annual visitors, 40% of whom are tourists. The Mall of America—the heart of American retail—is the site of our case study.

In collaboration with the Best Buy corporation, we installed a video recording system in one of the smaller shops in the mall, a Best Buy Mobile stand-alone store located on the first floor of the mall.[2] This store serves as a case study for the evaluation of the MIMIC model. This chapter describes the store and its operation, the video recording and tracking systems, and the preparation of the data for use by MIMIC to model and predict transactions. It also details some of the real-world challenges facing end-to-end systems like that of this thesis.

First, a cursory overview of the data pipeline.[3] Data from the case study comes from two sources: a corpus of multi-camera video captured from ceiling mounted cameras onsite, and the transactional record of purchases made at the store. Video is processed in several steps through a pipeline that terminates with the MIMIC prediction model. First, people are tracked within

[1] `http://www.mallofamerica.com/about/moa/facts`

[2] A full-sized big-box Best Buy store is located on the mall's third floor.

[3] Technical details about the recording and data pipeline can be found in Appendix A.

**Figure 4.1:** The Best Buy Mobile Mall of America Store. This photo collage was taken near the rear of the store. The store's entrance can be seen on the right-hand side. Four of the video cameras installed in the store are visible attached to panels on the ceiling.

the store, and those trajectories segregated into employees and customers. The trajectories are next filtered to remove systemic errors, smoothed, and merged between cameras. Finally, floc activity features are generated from the trajectories, and training and evaluation sets are created with the electronic transaction record. This curated real-world dataset is used to evaluate the Mimic customer model.

## 4.1 The Best Buy Mobile Stand-alone Store

Best Buy Mobile (BBM) is a separate business unit within the Best Buy corporation. BBM operates small retail stores with a product spread focused on their core competence of mobile phones and mobile computing. The case study store measures approximately 1850 square feet (171 $m^2$) in retail area and carries a variety of cell phone and laptop computer products, as well as a diverse array of accessories. The portable phones are divided among three of the four major carriers (Sprint, Verizon, and T-Mobile). Contractual obligations prevent the fourth major provider, AT&T, from being sold at this store. The store has a small staff of which one to three members are typically on-site during weekday opening hours and as many as seven during weekend hours. For Best Buy, this was considered a concept store, with greater emphasis on portable computing than in their traditional retail locations.

The store is deeper than it is wide, with a glass facade that faces the

interior of one of the hallways of the mall's first floor. A point-of-sale register is located at the front on the left as you enter. Both left and right walls of the store are devoted mostly to mobile device accessories with a few specialized kiosks displaying a particular brand or device. In the center of the store are four single-person lounge chairs and a small coffee table. The floor space is shared among several tables which carry display units of various cell phones and portable computers. Three smaller tables are devoted to the major cell-phone service providers, and a fourth shows smart-phone and customization options. Along the rear of the store are three additional points-of-sale, as well as a Geek-Squad help desk and small passageway to a publicly accessible restroom. A doorway at the rear also allows access to a back-of-house storage and staging area. Also at the rear of the store is a small play area for children, and a laser-engraver for device personalization services offered at the store. Figure 4.2 shows a plan-view of the store.

Discussions with the store's staff, as well as with executives within Best Buy gave a general sense of the customer's behavior. Customers primarily come to the store for cell phones and accessories. Oftentimes, their visits are exploratory; they come to learn about different models of phones and the features and plans offered by service providers. Customers who purchase cell-phone accessories are often more goal-directed: they come with the intent to purchase, even if they have not decided which of several options to buy before arrival. Finally, a significant portion of customers visit the store with no intent to purchase; they arrive as part of a group, accompanying friend of family, or circumstantially (from boredom, or, for example, when their partner is visiting another store nearby).

A significant portion of the sales at this Best Buy store, in quantity rather than revenue, are of cell-phone related accessories. Sixty percent of transactions contained mobile-phone accessories. In comparison, twenty-six percent of transactions included cell phone hardware from one of the three major cellular carriers sold at the store (Verizon, Sprint and T-Mobile). The graph in Figure 4.3 shows the categorical breakdown of purchases in the store during the duration of the study.

Best Buy's business goal—beyond the selling of goods and services—is to develop long-term relationships with customers. Significant emphasis is placed on employees educating customers about products, teaching how use the often complicated technology, and servicing and troubleshooting phones that have been problematic for customers. As is the case with many other retail stores, employees make contact with customers entering the store as soon as possible with a "Hello," or "May I help you?"

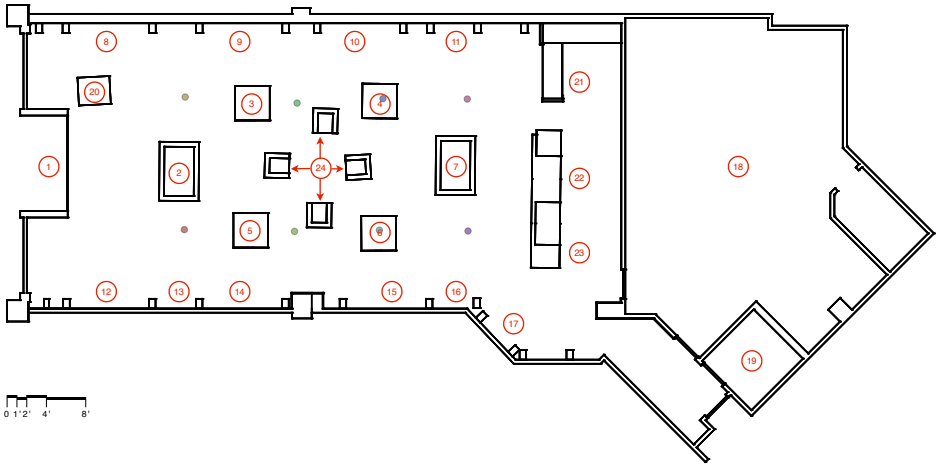

**Figure 4.2:** A plan of the Best Buy Mobile stand-alone store at the Mall of America.

| | |
|---|---|
| 1 | Entrance |
| 2 | Laptops |
| 3 | Sprint Phones |
| 4 | Verizon Phones |
| 5 | T-Mobile Phones |
| 6 | Smart Phones & Personalization |
| 7 | Laptops |
| 8 | No-contract Phones |
| 9 | Computer Accessories |
| 10 | Cases |
| 11 | Headphones & other accessories |
| 12 | Cases |
| 13 | Broadband |
| 14 | Cases & Shields |
| 15 | Chargers & Earpieces |
| 16 | Motorola Kiosk |
| 17 | Play area for children |
| 18 | Back of house |
| 19 | Public bathroom |
| 20, 22–23 | Points-of-sale |
| 21 | Geek Squad |
| 24 | Chairs |

The location of the eight cameras used for data-capture are marked in the image with colored dots. These locations were derived from camera calibrations.

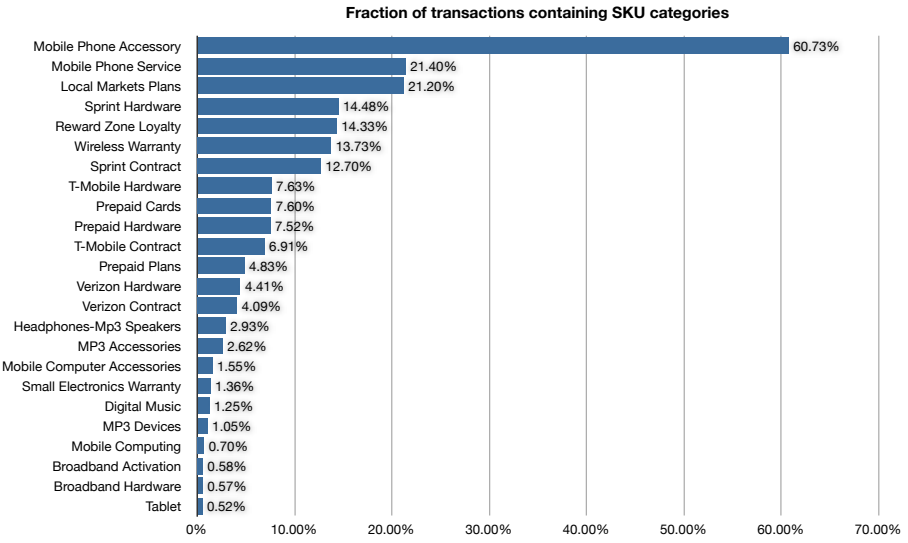


**Figure 4.3:** The fraction of transactions at the case-study store containing various product categories. Categories for each SKU present in a transaction were provided by Best Buy. This chart shows all categories present in at least 0.5% of transactions.

## 4.2 Recording at the Store

In the summer of 2010, we installed a video recording system in the store.[4] Each of the eight cameras installed captures images at one megapixel resolution (cropped to 960 by 960 pixels) at approximately fourteen frames per second. The images are encoded using JPEG compression, and pulled from the camera via IP networking. Custom recording software, developed for the Human Speechome Project (Roy et al., 2006; DeCamp, 2007), accumulates frames from all eight cameras, and records them to a hard disk array located in the back-of-house room of the store. Cameras are equipped with fish-eye lenses with approximately 180° field of view. Cameras timestamped each frame with microseconds since the epoch, UTC (January 1, 1970), and were synchronized via NTP.[5] Every few weeks, we shipped the disk array to the MIT Media Lab, where the recordings were transferred to a local disk array.

[4] This study was approved by the MIT IRB. "Retail Behavioral Pattern Analysis," COUHES application number 0809002886.

[5] Due to some errors with the on-site NTP server, the clocks on individual cameras began to drift for many of the recorded days. This required a manual resynchronization of data. The algorithm used to find individual camera delta-times is described in section A.3.

Unlike the Human Speechome recordings, no audio was captured at the Best Buy store. All processing of the video data—tracking, classification and prediction—was completed at the Media Lab.

We recorded video during the store's opening hours sporadically over the course of several months. Gaps in recording were due to either failure of the recording software or lack of sufficient onsite disk storage. Several days have partial recordings—video from fewer than all eight cameras; these recordings were not included in any of the modeling experiments. In total, 105 days were recorded, of which 75 days included video from all eight cameras (See Figure 4.5).

### 4.2.1 Electronic Transaction Records

Our collaborators at Best Buy provided a database of anonymized transactions for the recorded dates. Each record in the transaction database consists of fields for:

| | |
|---|---|
| **transaction ID** | A unique identifier for each transaction. |
| **timestamp** | in microseconds since the epoch, UTC. |
| **sku** | The stock keeping unit identifying a unique product. |
| **quantity** | The number of *SKU* items purchased (negative values indicate returns). |
| **register ID** | Which of the four points-of-sale were used for the given transaction. |

Additionally, Best Buy provided the hierarchy of internal categories used for each product. Each product (stock-keeping unit, or SKU) is a leaf in a category-tree that includes product subclass, class and department.

The mean number of transactions per day in the corpus was 32.29 (std. dev. 15.85), with a mean of 2.8 different SKUs per transaction. This translates roughly to a transaction every 4.75 minutes. Visual inspection of the video confirmed that the timestamps in the transactional record matched the timestamps of camera frames.

### 4.2.2 Camera Calibration and the Store Model

Electronic floor plans for the store were provided by Best Buy. Those plans, coupled with detailed measurements made on-site, were used to build a three-dimensional model of the store interior and key fixtures within. A global Euclidean coordinate frame was chosen in this model, and used during camera calibration. The origin of the coordinate system—a corner on the floor near the entrance—was chosen due to its visibility from several cameras. Custom

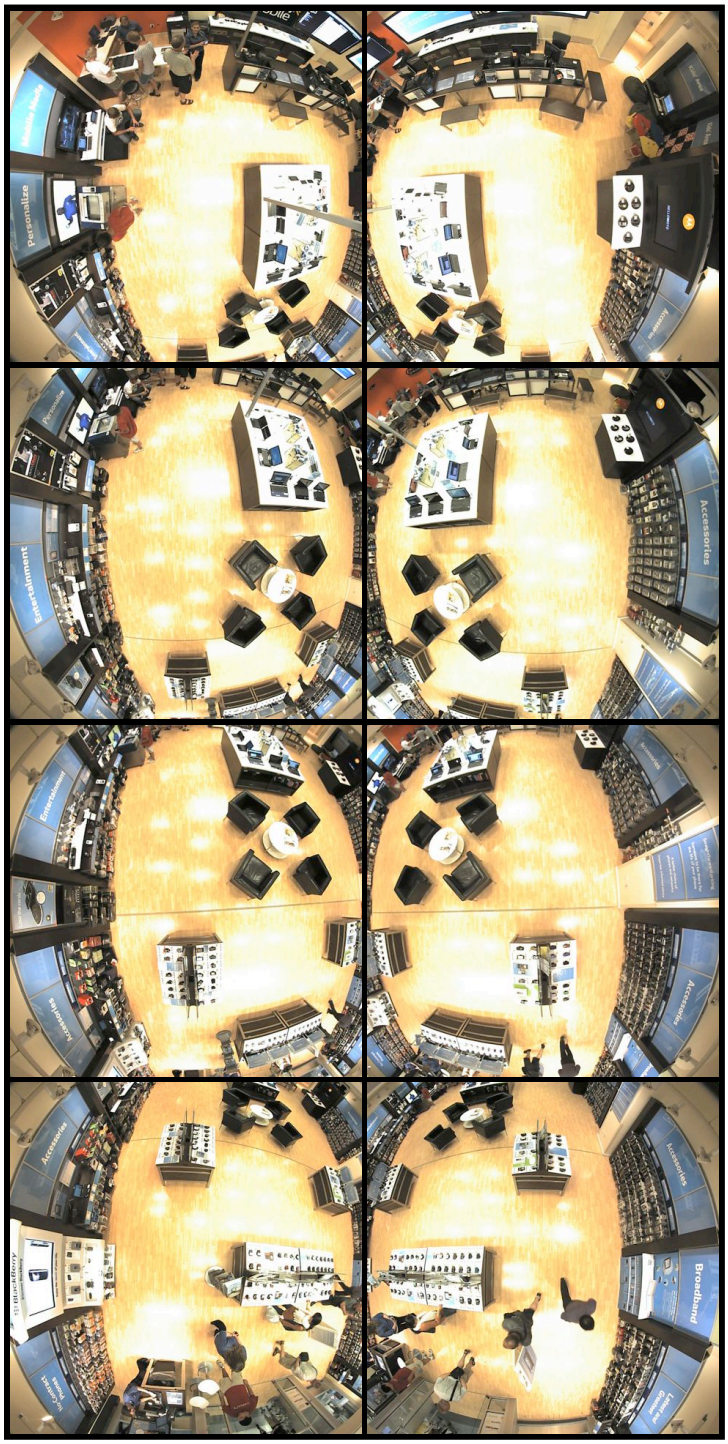


**Figure 4.4:** Images from the eight cameras installed in the Mall of America Best Buy Mobile store. The entrance to the store is on the left of this composite image. Each image is captured at 960 by 960 pixels, at approximately 14 f.p.s.

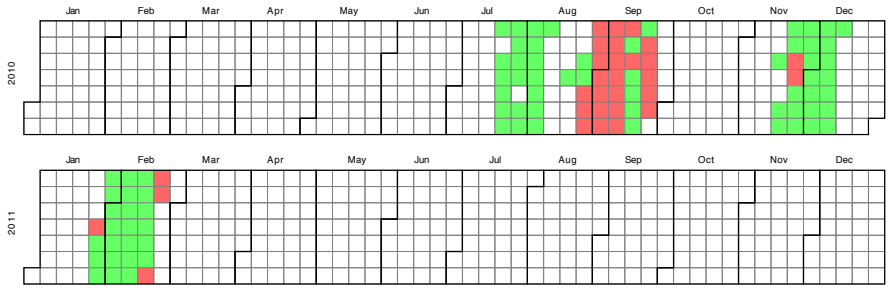


**Figure 4.5:** A calendar of days recorded at the Best Buy store. Days marked in green have data from all eight cameras. Days marked in red have partial data from less than all cameras. All modeling was done with data from days with complete recordings.

calibration software was used to calibrate both the location, orientation of each camera (its extrinsic parameters), as well as the parameters of a spheric model of the camera's lens (its intrinsic parameters). A detailed account of the lens model and parameter optimization can be found in DeCamp et al. (2010). Figure 4.6 shows a screenshot of the tool used to calibrate camera parameters.

The eight cameras were located at a height of approximately 3.1 meters off the floor (min: 3.03 meters, max: 3.21 meters).[6] Their height and locations afforded near-complete coverage of the public spaces in the store. Only the walkway behind the counter in the rear of the store, and the passageway near the public restroom were not fully covered. Each camera's field of view significantly overlapped with its neighbors. There was no video coverage of the exterior of the store (i.e. the windowed facade facing the inner gallery of the mall).

## 4.3 Tracking Customers

Pedestrians in the 5711 hours of recorded video data were tracked using a system named 2c, developed by George Shaw (Appendix A details the tracking algorithm. The system is also fully explained in Shaw, 2010, 2011). To speed tracking, video was first down-sampled to a resolution of 120 by 120 pixels. Tracks output by 2c each contain a camera ID, a set of $(x, y, t)$ tuples, and a collection of color histograms from the target person being tracked. Output $x$ and $y$ values are in image-coordinates; these are transformed into

[6] Camera heights are the values of their calibration's extrinsic parameters.

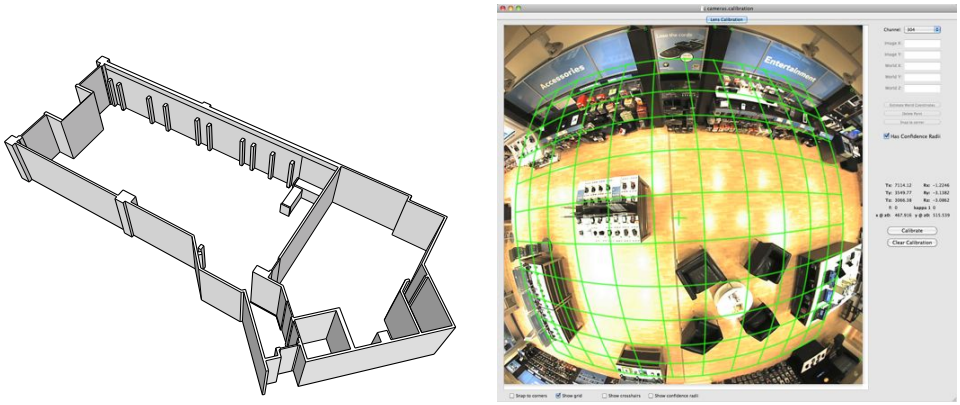


**Figure 4.6:** The 3D CAD model of the store, and a screenshot of the tool used to calibrate the intrinsic and extrinsic parameters of the cameras and lenses. The calibration tool is used to annotate the points in the image with their corresponding coordinates in the global Euclidean frame. Those global coordinates are read from the 3D CAD model. The grid displayed in this image represents squares one meter per side at the floor-level.

a global Euclidean coordinate space using the camera calibrations, then smoothed with a Kalman filter.

Using the camera's calibration, $\{x, y\}$ coordinates in an image can be mapped to a vector originating at the camera's location and extending in the direction targeted by the image coordinates. For a trajectory from a single camera, an approximation for the location of a person in the global Euclidean frame was calculated by finding the intersection of this vector and a plane located at a height one meter above the ground plane.[7] The choice of one meter was empirically motivated. To build a training set for the cross-camera merger (see Appendix B), I gathered a ground-truth set of tracks from different cameras tracking the same person. For each pair of these tied tracks, and for each frame of overlap between the tracks, I calculated the closest point equidistant from rays extending from each camera toward their targets. The average height of these points across all tracks and all annotated pairs was 1.088 meters.

The tracks generated using 2c undergo a series of post-processing filtering and consolidation steps. Very short tracks of duration less than a second or fewer than ten points—often the result of noise—are removed. Spurious tracks created from images on computer monitors and televisions in the store

[7] Metric units are used internally throughout all systems in described in this thesis. Imperial is used for descriptive purposes only.

are discarded using image masks. During several days of the video recording, balloons were posted at kiosks as part of a promotion. The movement of these balloons in ambient air currents caused a significant number of tracking false positives. Manually created image masks were also used for these days to remove these erroneous tracks.

Among tracking errors, the failure of the background-detection module of 2c is often the cause. In the Best Buy Mobile store, customers and employees will often stay still for several minutes at one spot to examine a product or process a transaction (cell phone activations often take in excess of twenty minutes)—long enough for the background-detection module to incorporate the customer or employee into the background model. When the people finally move, the true background is incorrectly labeled foreground, and a motionless track is created and sustained until the background model reabsorbs the region. Fortunately, this class of systematic tracking error is easy to detect and correct. Even when staying in the same location, the small movements of people in the video cause the tracker to occasionally shift its target a few pixels. The errors caused by a polluted background model can be detected and removed by looking for portions of a track where the target is at the exact same pixel for an extended period of time.

Another frequent type of tracking error results from occlusions. In the overhead fisheye views of the store, customers and employees often pass in front of each other; from the perspective of the 2c tracker, the two tracked objects appear as one. When this occurs, the 2c tracker will sometimes prematurely end one of the two tracks. When two people pass each other and create an occlusion, 2c will sporadically switch the targets of tracks. For example, if someone exits the store at the same time that another person enters, 2c may output a single track, rather than correctly registering two distinct tracks. This class of error has significant implications for downstream processing, especially with regards to cross-camera track merging. Advanced tracking algorithms which use more sophisticated object models would undoubtedly improve pedestrian tracking, the implications of which are discussed Chapter 6. The use of the floc activity vector mitigates this problem.

A final preprocessing step synchronizes track timestamps, adding an offset distinct for each camera. This synchronization step was necessary due to the intermittent failure of the NTP daemon on the on-site recording server. The algorithm used to generate time-offsets for each camera is described in Appendix A.

A person in the store generates simultaneous tracks from several cameras due to the overlap of fields-of-view of the eight cameras. Tracks targeting

the same person are clustered in a post-processing step described in detail in Appendix B. Clusters are then merged into a single trajectory in the global coordinate frame by averaging stereoscopic estimates from each pair of cameras which target the same person:

**Algorithm 1** Procedure for calculating a merged track in the global coordinate frame from a cluster of multiple trajectories.

$\tau = \{\tau_1, \ldots, \tau_k\}$ : tracks in cluster.
$t \leftarrow$ earliest start time in $\tau$
$t_e \leftarrow$ latest end time in $\tau$
**while** $t \leq t_e$ **do**
  **for all** track pairs $\{\tau_i, \tau_j\}$ which exist during $t$ **do**
    $l_i \leftarrow$ the line defined by the location of camera $i$ and the target image coordinates $\tau_i(t)$.
    $l_j \leftarrow$ the line defined by the location of camera $j$ and the target image coordinates $\tau_j(t)$.
    $p_{ij} \leftarrow$ the nearest point equidistant from $l_i$ and $l_j$. I.e. the point closest to intersection between the two lines.
  **end for**
  $\tau(t) \leftarrow$ mean of all points $p_{ij}$
  $t \leftarrow t + \Delta t$ {$\Delta t$ is a constant time increment.}
**end while**

With tracking and merging completed, the roughly 600 GB of daily video is reduced to approximately 700 MB of tracks. On average, each day produces 23.4 thousand tracks (std. dev. 16.8 thousand). Figure 4.7 shows an overlay onto a plan of the store a single day's tracks. Across the entire corpus, 41,245 GB of video was transformed into 68 GB of tracks.

### 4.3.1 Customer / Employee Classification

Employees at the Best Buy store wear a consistent uniform of black pants and dark blue shirt with a small yellow logo. On some days, a specialist employee worked with a black sports jacket or white shirt and black tie. The uniformity of their garb makes viable customer/employee classification using a track's associated color histograms.

Color histograms output during tracking were used to build a simple discriminative classifier to divide tracks between customers and employees. Histograms were sampled at the first and last frames of every track, and at regular intervals during the track. Color histograms consist of 512

**Figure 4.7:** Above: tracks from a short three minute timespan. Below: The filtered and merged tracks of a single day (August 8, 2010) in the Best Buy Mobile store. In this image, red tracks correspond to those tracks classified as customers; blue tracks to those classified as employees.

bins, uniformly divided across the RGB color-space (eight bins across each dimension).

The classifier is simple: Two color histograms are built based on training data—one for customers, the other for employees—by summing training histograms. The histogram for employees is very "peaky", with much of the mass centered on the dark grays and blues; that for customers is more uniformly distributed across the color spectrum. The color histograms, with a Laplacian smoothing, were used as estimates of the probability distribution of the colors of employees and customers. To classify a track, its aggregate color histogram—the sum of all histograms associated with the track—is compared against the two trained histograms. Whichever histogram better matches the candidate, as measured by lower Kullback-Leibler (KL) divergence, wins the label. Explicitly, if the histogram for the target track is $h_\tau$ and aggregate color histograms for the customers and employees are $h_{\text{cust}}$ and $h_{\text{empl}}$, respectively; where $h(i)$ is the probability of bin $i$:

$$\text{label}(\tau) = \begin{cases} \text{“customer”} & \text{if } \text{KL}(h_\tau, h_{\text{cust}}) < \text{KL}(h_\tau, h_{\text{empl}}) \\ \text{“employee”} & \text{otherwise} \end{cases}$$

where the KL divergence of two color histograms $h_1$ and $h_2$ is:

$$\text{KL}(h_1, h_2) = \sum_i h_1(i) \log \frac{h_1(i)}{h_2(i)}$$

The employee/customer classifier training set is comprised of the color histograms of 3456 manually labeled trajectories (1876 customers, 1580 employees). Table 4.1 lists details of the classifier's performance on a held out test set. Customers classified as employees (false positives) were customers whose attire too closely resembled employees. Employees classified as customers (false negatives) were typically of the employee wearing a black sports-coat.

Activity pattern features were generated using the tracks classified as customers and the functional locations described below.

## 4.4 Selection of Store Functional Locations

I defined thirty-nine functional locations for the Mall of America Best Buy, placed around distinct areas in the store. Flocs were chosen to divide product categorical boundaries as well as areas with distinct uses (for example, a small play area for children, or each lounge chair). Figure 4.8 shows the

| | Predict Pos. | Predict Neg. |
|---|---|---|
| Actual Pos. | 885 | 42 |
| Actual Neg. | 190 | 832 |

| Metric | |
|---|---|
| Accuracy | 0.881 |
| Precision | 0.955 |
| Recall | 0.823 |
| MCC[a] | 0.772 |

[a] Matthew's Correlation Coefficient

**Table 4.1:** Performance of the color-histogram based Employee/Customer track classifier. Here, labeling a candidate track "employee" is the positive case.

location and extent of flocs in the store. Points of sale were explicitly not chosen as functional locations. Their inclusion in transaction modeling would lead to a trivial result: since transactions are completed at registers, we could expect activity at those locations immediately before a transaction's timestamp.

Some functional locations, the central chairs in particular, were defined around furniture that can move. I observed by watching video from each recorded day that the location of these flocs remained relatively constant during the course of a day—employees would reposition furniture to its "default" location when moved. Over the course of several days, some pieces of furniture could be jostled and moved slightly to create a new "default" location. I built an annotation tool (see Figure 4.9) to locate furniture within the store and collected location data for each floc-associated piece of furniture for days in the dataset.

### 4.4.1 Store Layout Changes

During the period of data collection, there was one significant change to the store's physical layout. Initially, two smaller tables with cell phones were placed together at the front of the store. A second pair of tables were placed deeper inside, split to the left and right side. At the rear of the store, between the chairs and back-wall point-of-sale, two larger tables carrying laptop computers were placed together. Prior to the holiday shopping season, one of the two larger laptop-carrying tables was brought to the front of the store, and the pair of smaller cell-phone tables split and placed on either side of the store. Figure 4.10 shows the variants of the store configuration.

The thirty-nine flocs followed changes in the store configuration: functional locations target *function* rather than *location*. I.e. the floc encompass-

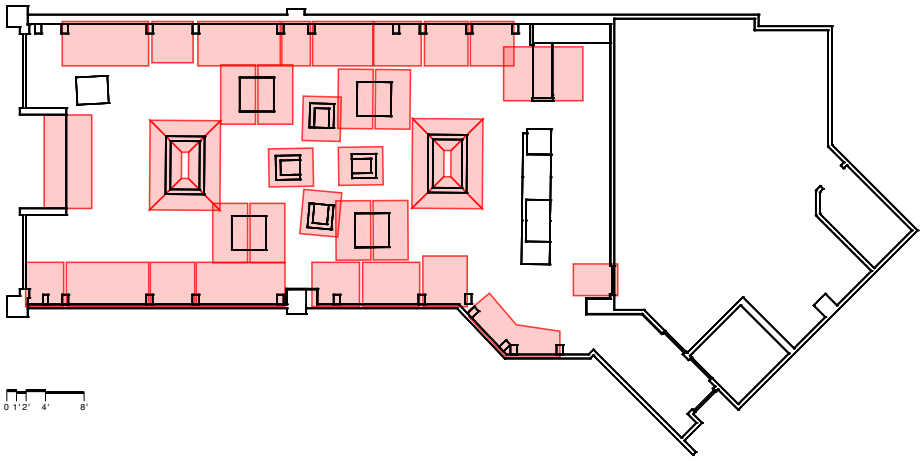

**Figure 4.8:** The layout of flocs within the store. Some flocs were attached to objects or locations that can move (for example, the chairs at the store's center). For those flocs, I annotated daily location changes. Note that points-of-sale were explicitly *not* chosen as flocs.

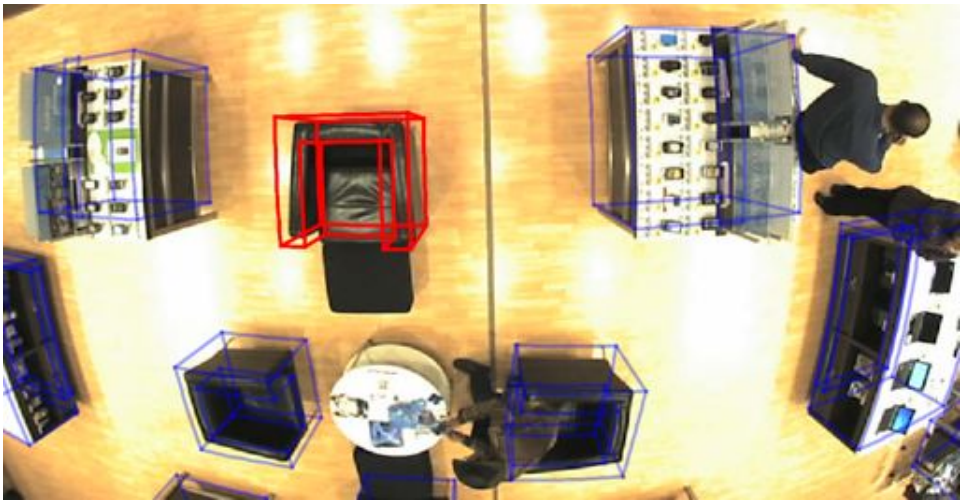

**Figure 4.9:** Screenshot from the furniture-placement annotation tool.

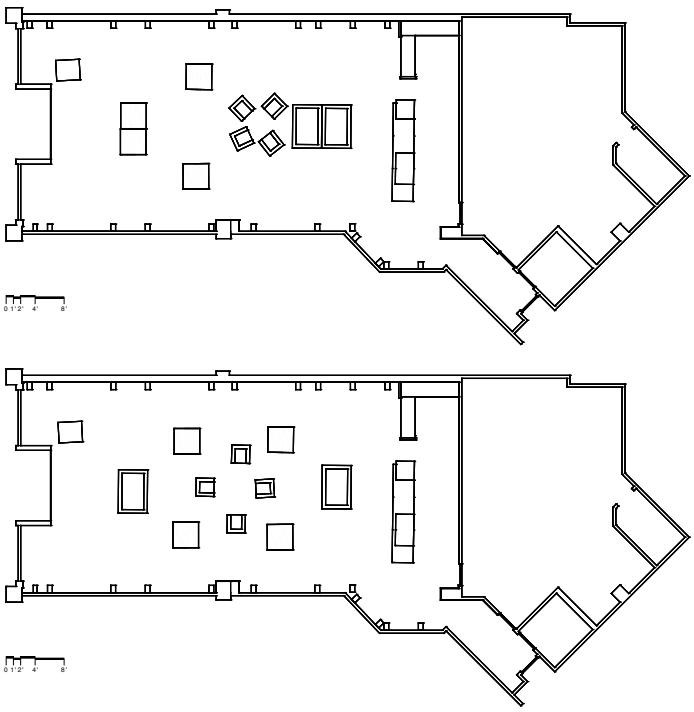

**Figure 4.10:** The layouts of the Best Buy store during the case study's data collection.

ing "sprint-phones" always encloses the kiosk displaying phones from that provider.

## 4.5 Data Preprocessing

The final step in the preparation of data for the Mimic model is the generation of floc activity vectors and the selection of positive and negative examples for training and testing. Activity vectors were generated for open hours of the store (9:00 AM–8:00 PM) with a feature timebase $\Delta t$ of five seconds. This timebase is long enough to capture meaningful interactions between customers, employees and products in high fidelity. Before being modeled by the HMMs of Mimic, the activity patterns are further temporally smoothed by convolution with a rectangular window. This smoothing is further described and explored in the following chapter. The decimation of track data into floc features further compresses the data; the 700 MB of a day's tracks is reduced to 2.4 MB of activity patterns.

## 4.6 Challenges in the Best Buy Mobile Store

Compared to many other types of retail stores, this Best Buy Mobile store poses particular challenges to predicting purchases. Unlike similar studies performed in stores vending staple goods, or an array of products liable to be purchased spontaneously (see for example Hui et al., 2009a; Kholod et al., 2010; Yada, 2011; Larson et al., 2005, which examined behavior in supermarkets), the high price-point of products at this store, and the contractual obligations demanded by cell-phone service providers, lead customers to make more deliberative choices. Customers will visit retailers several times to physically handle products and investigate different options before a purchase decision is made.

Moreover, there are a huge array of potential confounds. Promotions may drive one particular product's sales, and employees may have different skills closing a sale. These factors are divorced from the physical configuration of a space, and hence invisible in the data input into the Mimic model.

## 4.7 Conclusions

The Best Buy Mobile stand-alone store used as a case study to evaluate the Mimic model is a representative, if challenging, retail space to model. Identifying episodes of transactions within this class of store is enormously

difficult. Tracking pedestrians in the video is problematic due to frequent occlusions, long periods when customers and employees remain relatively still, and the compounding of errors when handing off trajectories from one camera to another. The type of products being sold within the store further complicate the classification task; due to the high cost of many of the items sold in this Best Buy, customers make less frequent impulse buys, and often come to the store to learn about the products only to return at a later date to make a purchase.

The retail store is not a controlled experimental environment where the physical configuration can be treated as an independent variable. There are huge fluctuations in the store's sales: temporal (the holiday shopping season, promotions), categorical (the iPhone is the most popular smart-phone), and social (efficacy of employee salesmanship). Each of these contributes to the challenges to modeling transactions.

The next chapter evaluates the performance of Mimic on the Best Buy Mobile Mall of America data and several subsets.

Chapter 5

# Performance of the MIMIC model

How well does the MIMIC model predict transactions? In this chapter, the floc activity pattern data gathered in the case study retail store is used to evaluate the model, both in its sensitivity to free parameters, and in contrast to several alternative models. Following a description of the training data used, parameter learning, and the metrics of evaluation, is an exploration of the space of free parameters of the model and a discussion of their effects on model performance. Next follow several experiments with the model, and a discussion of the findings.

## 5.1 Positive and Negative Training Sets

MIMIC requires a training set of labeled example episodes. As discussed in Chapter 3, a day's activity patterns is divided into positive and negative labeled examples by first labeling positive examples as all time-periods of duration $\psi$ preceding transactions, then dividing the remaining unclaimed data into $\psi$ length episodes with a negative label. The ratio of counts between positive and negative examples is determined by the value $\psi$. Ideally, the duration $\psi$ captures the typical, important period where purchase decisions are made.

Unless otherwise noted, models were evaluated using randomized five-fold cross-validation. A fold's training and test data were selected without replacement from the randomized complete set of positive and negative examples. This poses a complication due to the overlapping data used in positive examples. In a naïve cross-validation, examples in a cross-validation fold's test-set may contain data used to train that fold's model due to

the overlapping data present in some positive examples (see Figure 3.2), which could mask over-fitting of the model. To prevent artificially inflated performance results by cross-contamination, I removed from a fold's test-set any episodes which temporally overlap episodes in the fold's training data.

Randomized cross-validation obscures any potential effects of changing store layout, promotions, and seasonal variations (i.e. the Christmas shopping season). The individual effect of these confounds is difficult to disentangle; an experiment with the model investigating seasonal/layout effects is described in Section 5.9.3.

## 5.2 Parameter Learning & Implementation Notes

The general techniques used in learning the static and dynamic models of activity patterns—counting and expectation maximization (EM)—are described in Chapter 3; some implementation-specific aspects merit mention. For all models, log-likelihoods were used rather than direct probabilities for numeric accuracy and computational simplicity.

The Gaussian components in the GMM static model used only a diagonal covariance. This reduction in model expressivity can be mitigated by increasing the number of mixtures in the model. As mentioned in Chapter 3, a minimum variance was enforced along each dimension to prevent mixtures of single examples with zero variance and infinite maximum likelihood.

Expectation-maximization based parameter-fitting of both static and dynamic models terminated when the ratio of log-likelihoods between EM iterations was less than a threshold $\lambda$. If the likelihood of the data $x$ at iteration $i$ is $\mathcal{L}(x|\theta_i)$ then EM terminated when

$$\frac{\log \mathcal{L}(x|\theta_{i-1}) - \log \mathcal{L}(x|\theta_i)}{\log \mathcal{L}(x|\theta_{i-1})} < \lambda.$$

In all experiments presented here, $\lambda$ was set to 0.000001. I cross-validated a typical parameter setting five times with different random initialization. The AUC (area under the ROC curve) of the resulting models was near identical (standard deviation of 0.00313), evidence that the model does not terminate prematurely in a local optima.

All model parameters were initialized using a random subsampling of the training data.

Cross-validation of mimic models was performed on a heterogeneous

computing cluster of approximately fifty computing cores[1] which could complete several thousand parameter configurations during a weekend.

## 5.3 Metrics of Evaluation

Rather than use a single binary confusion matrix or derived characteristics such as accuracy, $f$-score or Matthew's Correlation Coefficient (MCC), to evaluate the performance of any single model parameter setting, I use the area-under-curve (AUC) of the receiver operating characteristic (ROC) curve. The AUC is a measure of the overall performance of a binary classifier as a parameter is varied (Davis and Goadrich, 2006). In the cases presented here, the model prior—classification likelihood-ratio threshold—was varied. The AUC ranges between zero and one. A perfect classifier has an AUC of one, and a classifier performing at chance has an AUC of 0.5.

An ROC curve compares the false-positive rate (on the x-axis) against the true-positive rate (on the y-axis); both rates can vary between zero and one. A perfect classifier yields a point at the top-left of the curve, indicating 100% true-positives and 0% false-positives—i.e. no errors. The curve is pinned at the bottom left—declaring all examples negative—and at the top right—declaring all examples positive. A random classifier, one that chooses the class label with a weighted coin-flip, yields an ROC curve along the diagonal as the probability is varied. This is the chance line; classifier ROC curves that are above the diagonal perform better than chance. Comparing two models is done against chance performance. For example, a model with an AUC of 0.7 is considered to perform twice as well as a model with an AUC of 0.6; i.e. $\frac{0.7-0.5}{0.6-0.5}$.

## 5.4 Data Normalization

Figure 5.1 shows a clear relationship between the amount of activity in the store and the propensity for a purchase. We would like the floc activity model to capture patterns deeper than this trivial correlation. To force the model to find activity patterns in the data divorced from the *total* activity, the datasets were normalized to have proportional numbers of examples. The normalization procedure was as follows: positive and negative examples were

[1] Many of the machines in the cluster are the desktop workstations of my colleagues. These machines were removed from the cluster during the daytime. The size of the cluster was also reduced for memory-intensive tasks. This cluster was also used in earlier steps in the data processing pipeline such as tracking.

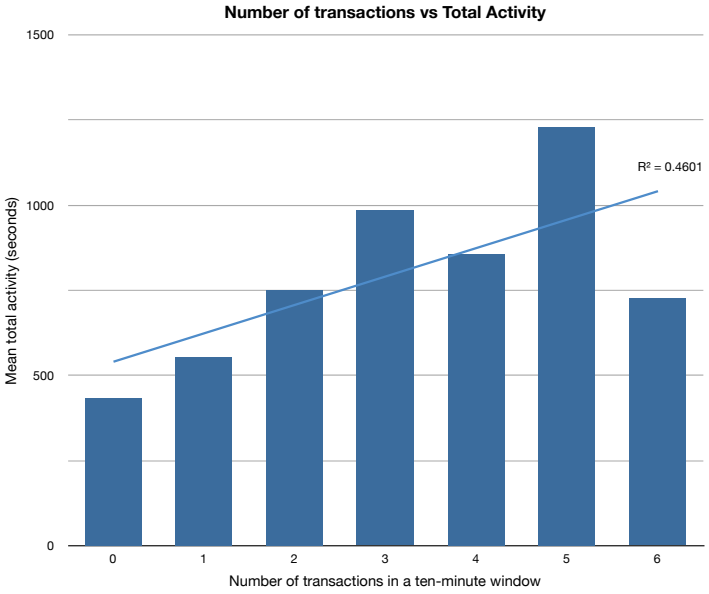


**Figure 5.1:** This graph shows the correlation between total activity during an episode and the number of transactions occurring during the episode. The graph was generated by sampling ten-minute episodes every five minutes, sorting the episodes by the number of transactions occurring during the episode, then calculating the mean of total activity (the sum of all activity vectors in the episode) per ordinal transaction count. A clear trend can be seen: the greater the amount of activity, the more likely a transaction is to take place. This makes intuitive sense: if every customer has a propensity for making a purchase, increasing the number of customers will increase the number of transactions that would be seen in a window of time.

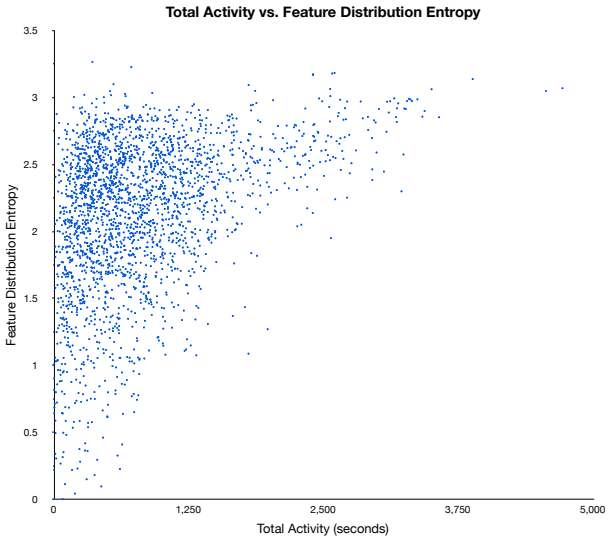


**Figure 5.2:** Graph showing the relationship between total activity and entropy of the activity pattern distribution, showing the trend of episodes with greater total activity having greater entropy as well.

binned by their total activity (the sum of all floc activity across all time samples). The bin size was 100 total seconds of activity. Examples were then drawn, without replacement, from each bin in fixed proportion between positive and negative examples. With this procedure, the distribution of total activity in both positive and negative training sets is identical.

An alternative normalization technique—scaling the features of every time-step so that they sum to one—is not a viable alternative to the method described above. At least, insofar as it does not completely remove information about the total amount of activity in an episode. Assume the true distribution of activity in the store is $D$. Every quantum of activity (i.e. the activity of a single person) at a moment in time is sampled from this true distribution. Every episode is made from a collection of many of these drawn samples. As the number of samples increases, the divergence between the true distribution and the sampled distribution goes to zero.

How could a classifier take advantage of this fact to predict the total activity? One possibility is to calculate the entropy of an episode's activity distribution; the lower the entropy, the lower the likely total activity. Figure 5.2 shows a plot of an episode's mean activity distribution entropy versus the total activity, revealing the clear correlation between the two factors.

| Parameter | Values |
|---|---|
| state count $z$ | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 |
| state model | binomial, multinomial, GMM |
| state graph topography | Simple-chain, Bi-directional chain, Skip-chain, Fully connected |
| mixture count (GMM only) | 1, 2, 3, 5, 10, 15 |
| episode duration $\psi$ | 10 minutes |
| smoothing window | 10, 50, 100 seconds |
| dataset | balanced & unbalanced |

**Table 5.1:** The cross-product of the above parameter settings was evaluated in the coarse-grained exploration of the Mimic parameter space.

| Parameter | Value |
|---|---|
| state model | Multinomial |
| multinomial $\delta$ | 1 microsecond |
| state count $z$ | 3 |
| state graph topography | Fully connected |
| episode duration $\psi$ | 10 minutes |
| smoothing window | 110 seconds |

**Table 5.2:** Parameters of the baseline Mimic model. The AUC of this model is 0.5822.

## 5.5 An Exploration of the Parameter Space

The full free parameter space of Mimic is huge: the cross-product of seven parameters. I performed a course-grained exploration of the full space,[2] evaluating over fifteen thousand different settings, with more fine-grained evaluations in several parts of the space. These explorations varied the seven model parameters described below (Table 5.1). Rather than an exhaustive summary of the results of this exploration, I present here the effects of the variation of several parameters as all other parameters are held constant. Unless otherwise noted the results reported here vary parameters relative to this baseline parameter setting, which I found yielded middle-of-the-road performance. Table 5.2 enumerates the parameters for the baseline model.

[2] Up to fifteen GMM mixtures and ten HMM states.

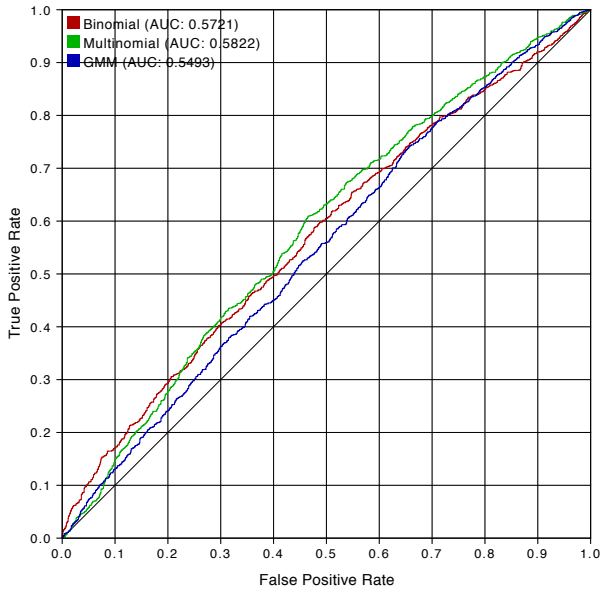


**Figure 5.3:** ROC curve for the three variants of static state models.

**State types** Model performance using the three static distribution models is shown in Figure 5.3. Here the AUC for static state models of type binomial, multinomial and GMM are 0.5721, 0.5822 and 0.5493, respectively. Compared to other model parameters such as state counts or example size for which one might expect smooth changes in performance as the parameter is varied, the choice of state model structures different performance manifolds. Although the multinomial static model bests alternatives relative to the baseline parameters, this result does not generalize to the best-performing Mimic models (see Table 5.3). In general, the GMM-based classifiers were outperformed by both multinomial-state and binomial-state variants with comparable parameter settings. Building an intuition that explains these variants is difficult, especially when the rank order of model performances change as other parameters are varied (for example, with the best performing models).

**HMM state counts** The number of states in the HMM is associated with the complexity of temporal patterns that can be modeled. If there exists common subsequences of activity patterns in the data, that we expect that models with more states would be better performing. This is not the case.

Independent of state model, the best performing classifiers had few states, with little variation in performance as state-counts are increased (Figure 5.4).

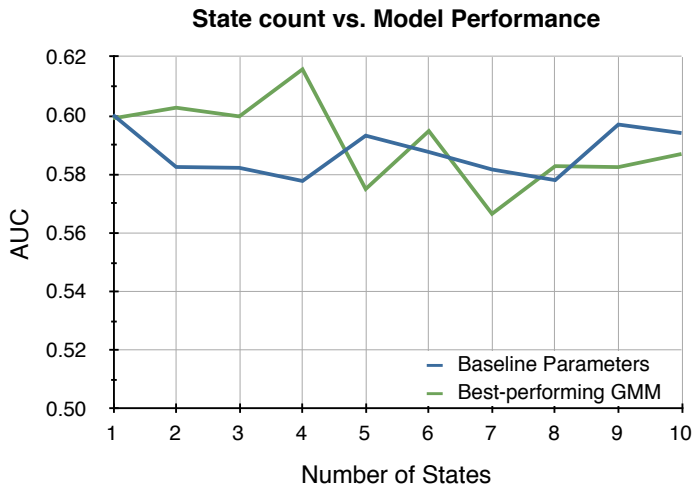


**Figure 5.4:** The performance of the model as the number of states is varied. Here is shown the AUC of the fixed baseline model and best-performing GMM-based model as the number of states ranges from one to ten.

This indicates that—at the resolution afforded by the activity vector features—there is little dynamic structure that MIMIC can capture, especially when using the less descriptive binomial and multinomial state models. Variations in performance are more prominent with the Gaussian mixtures state model. The best performing GMM-based MIMIC model on the balanced dataset uses four states (nine on the unbalanced dataset).

**Example size** What is the best window of activity one should examine when predicting transactions? Too short a window may not include activity patterns indicative of a transaction; too long may stretch before a purchasing customer has ever entered the store. We might expect that the best performing MIMIC model would use an example size equal to the mean time spent in the store by customers who make purchases.

Surprisingly, increasing $\psi$ had the effect of increasing classifier performance. Figure 5.5 shows the variations of performance of the baseline parameterized model when $\psi$ is varies from two minutes to forty-four minutes. Why would performance continue to increase, even for very long episode lengths? One possibility is that the longer episodes better capture transactions which include sales and service of new cell-phone plans. This class of transactions often takes thirty minutes or longer as employees register the phone, perform credit checks and educate the customer about their purchase.

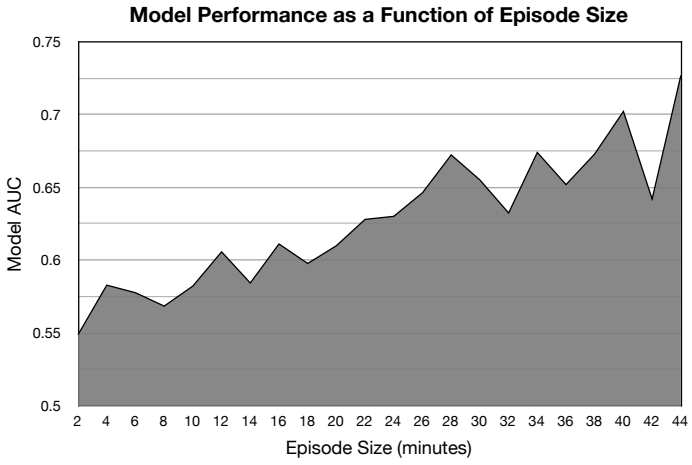


**Figure 5.5:** Performance of the baseline model as a function of episode duration $\psi$.

The increasing performance result may also be an artifact of the evaluation process. As we have already seen, the probability of a transaction occurring increases with the total amount of activity in the store. Also, any episodes which temporally overlap with data in the training set are excluded from the cross-validation test set. Taken together, these points imply that as the episode duration increases, more positive examples will be systematically excluded from the test set, artificially increasing the reported performance of classifiers that perform well on low-activity (and thus more likely negative) episodes.

**HMM graph topology** We would expect the topology that matches the dynamics of the true underlying distribution to be best-performing if the model training terminates prematurely. For example, if activity in the store typically transitioned through a sequence of states representing *browsing*, *inquiry*, and *pre-purchase*, but infrequently between *browsing* and *pre-purchase* states, then we could anticipate the fully connected graph topology to underperform. The fully-connected topology is a superset of the constrained variants and can fully model these more restricted sequences. If the fully-connected topology is most performant, then either there does not exist a clear underlying temporal structure to the data, or the EM algorithm does not enter a lull in model improvement.

The fully-connected state graph topology is the most expressive of the four variants, and evaluations of the model variants bear out the supposition that the fully-connected HMM would perform best, if only slightly. Figure 5.6 shows a typical pattern where the fully-connected model performs

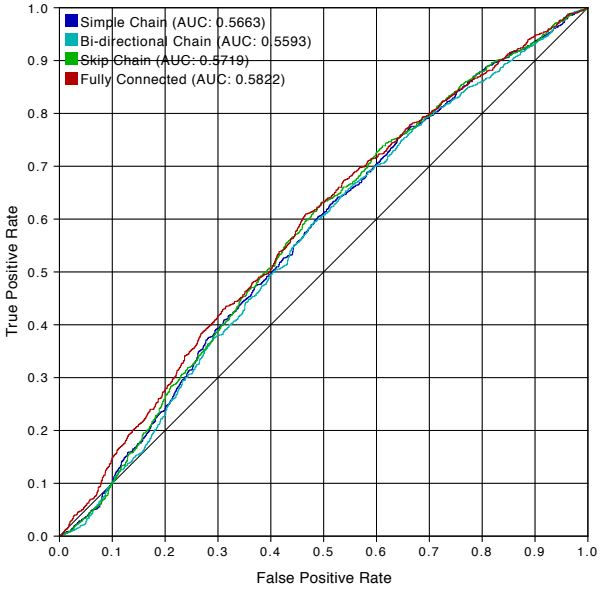


**Figure 5.6:** ROC curve for the HMM state graph topology. Shown are simple-chain, bi-directional chain, skip-chain and fully-connected.

approximately 24% better than the simple-chain topology, compared to chance.

The best-performing binomial models are single-stated (i.e. they revert to the static model) and therefore graph topology is irrelevant. Likewise, the best-performing multinomial models use two states, so only the fully-connected and simple-chain variants apply—of that pair, the fully-connected models perform best.

**Data smoothing** One of the common characteristics observed in the data are what might be called "islands of activity": A customer browsing items in the store spends several seconds in transit walking from one floc to another, leaving no trace (i.e. zeros) in the floc activity data. These isolated activities can be harder for an HMM to model.

To connect these islands, I smoothed the data by convolving it with a rectangular kernel. Once smoothed, I subsampled the data at fixed time increments half the width of the kernel. An episode $V$ is smoothed with a

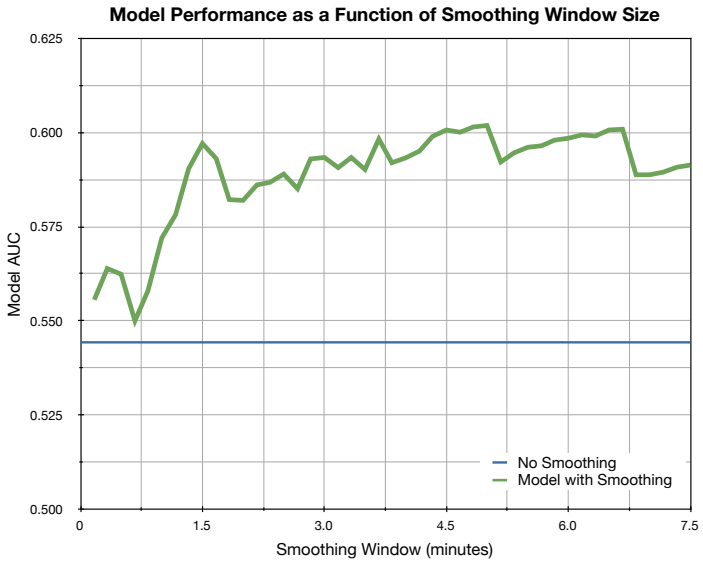


**Figure 5.7:** The effect of data smoothing.

kernel of $q$ samples, and resampled to become $V''$:

$$V' = \{\vartheta'_1, \ldots, \vartheta'_m\}$$
$$\vartheta'_i = \sum_{j=i}^{i+q} \vartheta_j$$
$$\vartheta''_i = \vartheta'_{q \cdot i}$$

Subsampling the smoothed data has the additional effect of reducing computation during training and evaluation.

Smoothing significantly improves classifier performance, with diminishing returns beyond a smoothing window of approximately two minutes (See Figure 5.7). To continue the analogy, with smoothing the activity "islands" are joined with each other, allowing the HMM to more easily hop between them. With the baseline parameters, Mimic performs twice as well when data is smoothed with a 2.5-minute kernel compared to without smoothing.

| Parameter | Balanced dataset | | |
|---|---|---|---|
| State model | Binomial | Multinomial | GMM |
| State count $z$ | 1 | 2 | 4 |
| State graph topography | — | Fully connected | Fully connected |
| Mixture count | — | — | 15 |
| Smoothing window (seconds) | 110 | 220 | 110 |
| AUC | 0.6243 | 0.6158 | 0.6070 |

| Parameter | Unbalanced dataset | | |
|---|---|---|---|
| State model | Binomial | Multinomial | GMM |
| State count $z$ | 1 | 2 | 9 |
| State graph topography | — | Fully connected | Fully connected |
| Mixture count | — | — | 10 |
| Smoothing window (seconds) | 100 | 10 | 110 |
| AUC | 0.7400 | 0.6740 | 0.7119 |

**Table 5.3:** Parameter settings for the top three Mimic models in both balanced and unbalanced datasets. The episode duration ($\psi$) for each of these models was fixed at ten minutes.

## 5.6 Best-performing Models

Of the thousands of parameter variants evaluated, the settings of the best performing binomial, multinomial and GMM-based models are listed in Table 5.3. On the unbalanced dataset, the binomial static model was the best performer, with an AUC of 0.6243. The binomial also bested the alternatives on the unbalanced data, with an AUC of 0.7400.

### 5.6.1 Exploding a Model

By inspecting one of the better performing models, we can gain some insight into the kinds of activity patterns—and customer behaviors—that the model captures. The best-performing multinomial-based model contains two states with initial and transition probabilities shown in Tables 5.4 and 5.5 and visualized in Figure 5.8.

Both states in the positive model are nearly identical, with much of the probability mass distributed at one of the laptop tables near the entrance, and at the three smaller tables holding phones from the three cell-phone


**Figure 5.8:** The static distributions of states $A$ and $B$ in the best-performing multinomial model.

| State | Prior probability Positive | Negative |
|---|---|---|
| $A$ | 0.4993 | 0.1143 |
| $B$ | 0.5007 | 0.8857 |

| Positive Model | | | |
|---|---|---|---|
| | | Initial | |
| | | $A$ | $B$ |
| Next | $A$ | 0.5242 | 0.4657 |
| | $B$ | 0.4758 | 0.5343 |

| Negative Model | | | |
|---|---|---|---|
| | | Initial | |
| | | $A$ | $B$ |
| Next | $A$ | 0.9988 | 0.0002 |
| | $B$ | 0.0012 | 0.9998 |

**Table 5.4:** The learned priors and transition probabilities of the best-performing multinomial-based Mimic classifier.

providers. Also with a relatively high probability mass is the Geek Squad help desk, which perhaps indicates that Best Buy employees are successfully transitioning customer's problems into sales opportunities. Compared to the negative model's states, those of the positive place greater relative emphasis on the accessories on both sides of the store.

The two states of the negative model are more distinct from each other. The first state, $A$, concentrates most of its probability mass on the two Verizon flocs. In early January, 2010, Verizon officially announced that the popular Apple iPhone would be available on its network, with an availability of February 10, 2010.[3] Rumors of the iPhone's availability persisted for month before the announcement. The high probability on Verizon flocs may be explained by customers investigating Verizon service plans with no immediate intent to purchase, instead waiting for the iPhone to be available.

The second state, $B$, places much of the probability mass at the front laptop table. This state may capture relatively quiet moments in the store, where a few customers—perhaps window-shoppers—enter briefly to peruse laptops before exiting. Both states in the negative model infrequently transition between one another, indicating that store activity patterns remain comparatively constant for the ten minutes of an episode's duration.

[3] "Battle is Set as Verizon Adds iPhone," Jenna Wortham, New York Times, January 11, 2010. `http://www.nytimes.com/2011/01/12/technology/12phone.htm`

| | Positive Model | | Negative Model | |
|---|---|---|---|---|
| Floc | $A : P(\cdot)$ | $B : P(\cdot)$ | $A : P(\cdot)$ | $B : P(\cdot)$ |
| Entrance | 0.008544 | 0.008542 | 0.008159 | 0.007942 |
| Laptop-a1 | 0.017840 | 0.017841 | 0.010319 | 0.024980 |
| Laptop-a2 | 0.022877 | 0.022879 | 0.013008 | 0.024851 |
| Laptop-a3 | 0.102910 | 0.103039 | 0.045951 | 0.149602 |
| Laptop-a4 | 0.083059 | 0.083043 | 0.047660 | 0.113178 |
| Laptop-b1 | 0.008805 | 0.008795 | 0.008296 | 0.009640 |
| Laptop-b2 | 0.014688 | 0.014681 | 0.016124 | 0.013374 |
| Laptop-b3 | 0.012095 | 0.012083 | 0.006793 | 0.011911 |
| Laptop-b4 | 0.039839 | 0.039793 | 0.019968 | 0.047890 |
| Samsung | 0.000305 | 0.000301 | 0.001022 | 0.000199 |
| Cases-1 | 0.026166 | 0.026190 | 0.015350 | 0.017466 |
| Broadband | 0.013193 | 0.013194 | 0.007111 | 0.008471 |
| Cases/shields | 0.031847 | 0.031865 | 0.018358 | 0.021944 |
| Chargers | 0.010495 | 0.010499 | 0.010296 | 0.006632 |
| Earpieces | 0.007682 | 0.007679 | 0.008806 | 0.006956 |
| Motorola-1 | 0.008178 | 0.008176 | 0.010066 | 0.006896 |
| Sprint-1 | 0.066465 | 0.066483 | 0.035167 | 0.078602 |
| Sprint-2 | 0.079296 | 0.079356 | 0.027603 | 0.082009 |
| T-Mobile-1 | 0.053787 | 0.053813 | 0.040613 | 0.044920 |
| T-Mobile-2 | 0.044753 | 0.044762 | 0.028211 | 0.052573 |
| Verizon-1 | 0.049072 | 0.048942 | 0.112500 | 0.028317 |
| Verizon-2 | 0.044267 | 0.044196 | 0.148521 | 0.024665 |
| Smartphone | 0.022944 | 0.022952 | 0.032417 | 0.018050 |
| Personalization | 0.025622 | 0.025619 | 0.034580 | 0.022231 |
| Chair-1 | 0.016323 | 0.016297 | 0.064111 | 0.010595 |
| Chair-2 | 0.011654 | 0.011629 | 0.024510 | 0.008787 |
| Chair-3 | 0.013957 | 0.013917 | 0.027393 | 0.008590 |
| Chair-4 | 0.012623 | 0.012618 | 0.068312 | 0.010068 |
| Geeksquad | 0.042825 | 0.042977 | 0.017273 | 0.047931 |
| No contract | 0.016469 | 0.016485 | 0.014930 | 0.014651 |
| Motorola-2 | 0.008170 | 0.008162 | 0.004789 | 0.007151 |
| Comp.-Accessories | 0.030254 | 0.030237 | 0.026429 | 0.035081 |
| Wireless-Power | 0.006472 | 0.006458 | 0.003721 | 0.004258 |
| Cases-2 | 0.028775 | 0.028774 | 0.015680 | 0.017164 |
| Headphones | 0.007840 | 0.007827 | 0.005963 | 0.005222 |
| Etc. accessories1 | 0.005803 | 0.005804 | 0.004871 | 0.004110 |
| Etc. accessories2 | 0.002817 | 0.002815 | 0.002031 | 0.001595 |
| Play Area | 0.000338 | 0.000333 | 0.003199 | 0.000339 |
| Backroom door | 0.000950 | 0.000945 | 0.009886 | 0.001159 |

**Table 5.5:** The learned floc probability parameters of the best-performing multinomial-based Mimic classifier.

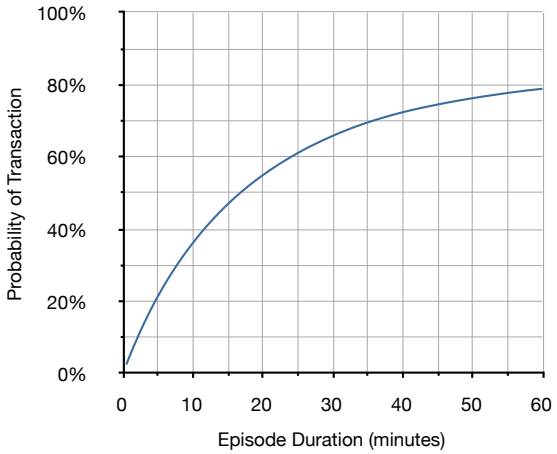


**Figure 5.9:** The probability an episode contains a transaction as a function of the episode's duration.

## 5.7 Alternative Models

This section compares the Mimic model against several competitors in order to give grounding to the model's performance.

### 5.7.1 Random Chance

The most trivial comparison is against random chance. Suppose a window of activity of duration $\gamma$ is chosen at random during the store's opening hours; what is the probability that the window will encompass a transaction? This probability is clearly a function of the duration of the window, and can be calculated directly from the data (see Figure 5.9).

As the duration of the window increases, so does the probability that the window will contain a transaction. For the ten minute window size used in most of the evaluations run here, the probability of a transaction in a randomly chosen episode is 36.2%.

### 5.7.2 Activity Thresholding

As noted earlier, there is a strong correlation between the total amount of activity in the store and the propensity for a transaction taking place. An extremely simple model can be constructed by comparing the total activity in flocs in an example against a fixed threshold. If the activity exceeds this threshold, the example is labeled as "transaction", and otherwise "no-transaction." The ROC curve shown in Figure 5.10 was created by varying this threshold. This trivial classifier shows reasonable performance with an

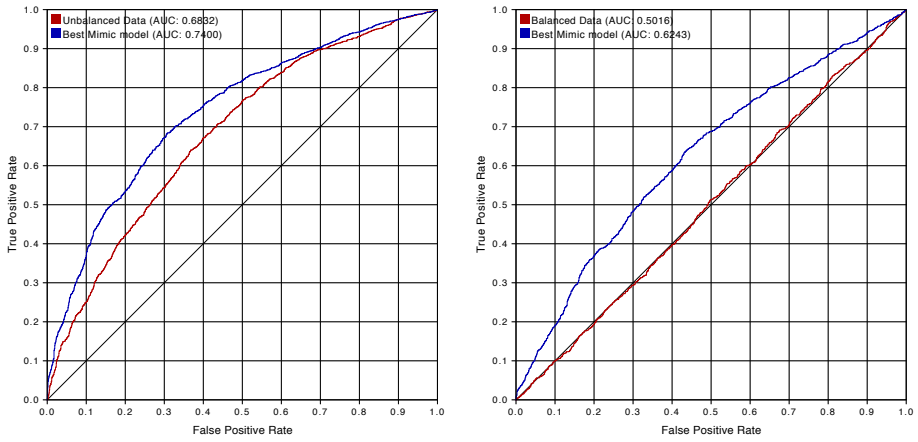


**Figure 5.10:** A trivial activity-based classifier compared to the best performing Mimic model. The ROC curves on the left were evaluated on unnormalized data; those on the right were evaluated with normalized data. Taken together, these graphs show that the Mimic model is capturing patterns more significant that sheer quantity of activity.

AUC of 0.6832, when evaluated with unnormalized data. The best performing Mimic model improves on the activity-thresholding model by 31% relative to chance. As expected, when the trivial thresholding classifier is applied to the normalized dataset, its performance is indistinguishable from chance, validating that examples are correctly normalized and that transactions can no longer be predicted by total activity alone.

### 5.7.3 Naïve-Bayes

A Naïve-Bayes classifier serves as a baseline to evaluate Mimic performance. An episode $V$, with $n$ flocs and $m$ activity vectors, is unravelled into a single feature vector of dimensionality $mn$. The grossly simplifying assumption made in the Naïve-Bayes model is that the activity at each moment in time is independent of all others—a hypothesis that is clearly not true with this data.

I used Naïve-Bayes implementation of the RapidMiner machine learning toolkit to evaluate this baseline model (Mierswa et al., 2006). The Naïve-Bayes classifier was evaluated with five-fold cross-validation with temporally-overlapping test-set examples removed. On the balanced dataset, the Naïve-Bayes classifier achieved an AUC of 0.5520; on unbalanced data, it achieved an AUC of 0.5685.

### 5.7.4 Transformations of the Data

Neither the activity-thresholding nor Naïve Bayes classifiers described above captures the temporal dynamics of the store's activity. Transformations of the raw data may reveal temporal patterns that could be exploited by a classifier to improve its performance. To test this hypothesis, I transformed the raw floc activity data using the discrete cosine transform and the Fast Fourier Transform (using only the real component of the result), both of which illuminate cyclic patterns by calculating the magnitude of sinusoidal components into which a signal may be decomposed. The transformation was performed independently on each dimension of the episode, and the transformed episode was flattened into a single vector.

So, for episode $V = \{\vartheta_1, \ldots, \vartheta_m\}$ where the number of floc features is $n$, the transformed vector becomes (in the case of the discrete cosine transform):

$$T(V) = \left\{\mathrm{DCT}(v_1^1, \ldots, v_m^1), \mathrm{DCT}(v_1^2, \ldots, v_m^2), \ldots, \mathrm{DCT}(v_1^n, \ldots, v_m^n)\right\}$$

The input vector was padded with zeros in order to expand the number of time-steps to be a power of two—a requirement for the Fast Fourier Transform.

The performance of the Naïve-Bayes classifier on FFT and DCT transformations of the data was identical for unbalanced data (AUC of 0.5989), and very similar on balanced data (AUC of 0.5619 and 0.5712 respectively).

### 5.7.5 State-of-the-art Black-box Models

The Google Prediction API[4] (GPA) is a black-box machine-learning web-service providing classification and regression of arbitrarily-dimensioned data (Google, 2011). Google Predict is a supervised system; users provide labeled examples to train a black-box model, and can later submit queries to the trained model. For classification models the query response contains label as well as label confidence values varying between zero and one. The system is black-box in that the machine-learning algorithms used by Google are proprietary—the public has no knowledge of the mechanism used to predict classes or regress values.

I submitted two variants of the floc activity data to the GPA: a flattened dataset identical to the dataset fed to the Naïve-Bayes model, and a dataset converted by the discrete cosine transform. I performed a five-fold

[4] Application Programming Interface.

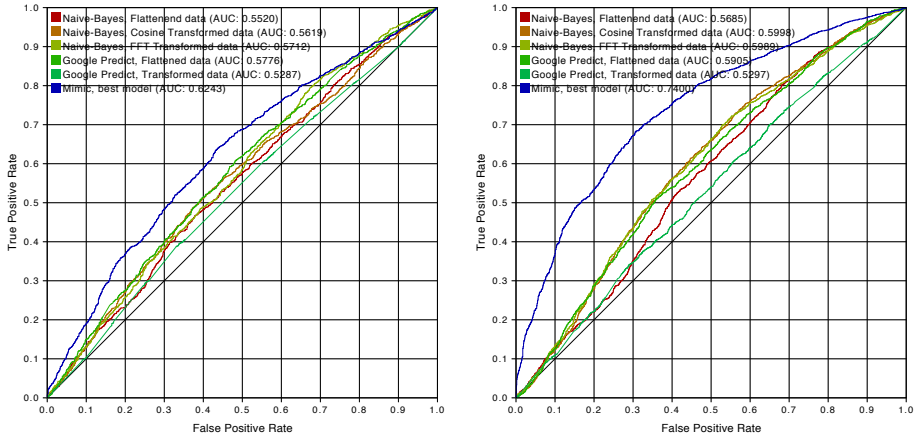


**Figure 5.11:** The performance of the best performing Mimic model vs several alternative models. On the left is the performance using the total-activity balanced dataset; on the right are the same models, trained and evaluated on the unbalanced datasets.

randomized cross-validation with the GPA model,[5] both on normalized and non-normalized data. The classification provided by the GPA model was discarded in favor of the class-confidence values—classification was instead made by comparing the difference in confidences against a threshold.[6] ROC curves for the GPA classifier were created be varying this classification threshold. Figure 5.11 shows the resulting ROC curves of the GPA and Naïve-Bayes models as well as the best performing Mimic model. Mimic is the clear victor in this contest, outperforming the GPA model by a factor of 1.6 compared to chance on balanced data, 2.7 compared to chance on unbalanced data.

## 5.8 Performance in Context

For those familiar with contemporary machine learning classifiers, the AUC performance of Mimic may seem underwhelming, especially with those tests run with balanced training data. But what is the best performance one could hope to achieve? Mimic attacks a problem where there may be very little

[5] As with the cross-validation used to evaluate the Mimic model, any examples in the test set which overlapped with data from the training were removed.

[6] The label given by the GPA is equivalent to a threshold value of zero; in effect, no information was discarded.

signal in a sea of noise. How much information can be extracted about an *individual's* choice from data derived from the *aggregate behavior of many*?

What if I brought my entire cognitive capacity to the problem? Could a human do better that Mimic? I generated a few hundred short video clips of episodes from the balanced sets. Each clip showed a plan of the store with the outline of the flocs within; the flocs lit up proportional to the amount of activity within. Viewing these videos, I was unable to determine which clips were generated from positive episodes, and which from negative, even with my intimate knowledge of the problem and its domain. This anecdote is far from definitive, but nevertheless illustrative of the challenges faced by any classifier making choices from activity patterns.

Mimic is useful despite performance slightly better than chance. Critically, by shifting the burden of evidence with the threshold $\gamma$, one can gain confidence in the classification of a subset of episodes. This fact enables the creation of several tools. These tools are discussed in greater detail in the following chapter, but the several experiments that follow show some additional ways Mimic may be harnessed.

## 5.9 Experiments and Explorations

The following experiments examine several additional factors that influence the performance of the mimic model.

### 5.9.1 Product Categories

Are some products or product categories more sensitive to patterns of activity than others? We can answer this question by selecting positive training examples filtered by the presence of a product category in a transaction; negative examples are chosen as in the standard model. Product categories were defined by the SKU taxonomy provided by Best Buy. In the case-study corpus, only sixteen product classes were present in fifty or more transactions. For each of these sixteen product categories, I cross-validated five Mimic models (varying state count between one and five, using the parameter settings of the best-performing binomial-based model for all other parameter values) on an activity-balanced dataset. Of the five models tested, the AUC of the best performer is shown in Table 5.6.

Episodes from several products categories proved more readily classifiable. Prepaid phones and related classes saw the greatest lift in performance, up nearly seventy percent relative to chance from the best performing standard model. Prepaid phones are physically located immediately to the left as one

| Category | # Ex. | Model AUC | Improvement |
|---|---|---|---|
| **Prepaid Plans** | 135 | 0.7099 | 68.87 % |
| **Prepaid Hardware** | 197 | 0.7095 | 68.54 % |
| **Headphones / MP3 Speakers** | 90 | 0.6735 | 39.58 % |
| **Prepaid Cards** | 185 | 0.6631 | 31.21 % |
| **MP3 Accessories** | 63 | 0.6586 | 27.59 % |
| **Verizon Hardware** | 112 | 0.6374 | 10.54 % |
| **Mobile Phone Accessories** | 1465 | 0.6340 | 7.80 % |
| T-Mobile Contract | 147 | 0.6191 | -4.18 % |
| Reward Zone Loyalty | 332 | 0.6155 | -7.08 % |
| Mobile Phone Service | 518 | 0.6101 | -11.42 % |
| Wireless Warranty | 347 | 0.6064 | -14.40 % |
| Local Markets Plans | 507 | 0.5981 | -21.08 % |
| Sprint Hardware | 359 | 0.5968 | -22.12 % |
| Sprint Contract | 316 | 0.5872 | -29.85 % |
| T-Mobile Hardware | 167 | 0.5843 | -32.18 % |
| Verizon Contract | 110 | 0.5431 | -65.33 % |

**Table 5.6:** Performance of the Mimic model when trained on positive examples segregated by product category. Those categories for which the model performed better than the best standard (unsegregated) model are marked in bold. Experiments were run using a balanced dataset. Improvement is relative to chance compared to the standard model. The AUC of the standard model is 0.6243.

enters the store (in the floor plans shown here, at the top left corner), behind a high desk on which sits a point-of-sale. Although the space is publicly accessible, I believe that the POS creates a psychological barrier that hampers customer exploration of the display area; as a result, customers that do make their way to the wall containing these products are more directed. Moreover, prepaid phones have a lower barrier to purchase than subscription-based plans which require a credit check and several-year commitment. The other product categories that saw performance improvement, mobile phone accessories, headphones and MP3 accessories also are inexpensive and have low barrier to purchase. A simple way to test this theory would be to physically swap the locations of prepaid phones with another product category, and observe if there is a subsequent change to the model performance.

### 5.9.2 Floc Sensitivity

What is the predictive power of any particular functional location? We can answer this question by reframing it as: How would model performance change if any particular functional location was not present in the data? I reran the three best-performing parameter settings for Mimic on floc activity data with each floc systematically excluded. By examining the change in AUC between the performance of the model with and without the inclusion of a floc, I derived a ranking of the relative importance of each individual location to model predictive power. The graph in Figure 5.12 shows these results ranked by the average change in model AUC. Negative values (i.e. worse performance without the floc) indicate that the floc is useful as a discriminator for purchase behavior; conversely, positive values mean that the floc confuses the model.

The results of this test show that the multinomial and GMM-based static models leverage every functional location to improve performance. Moreover, the ranking of floc values are roughly similar between the two model classes. Not so with the binomial static model; changes due to the removal of any particular floc were far less in magnitude and the rank ordering is scrambled compared to the multinomial and GMM orderings.

Among the three cellphone providers, flocs associated with T-Mobile were far more more indicative of intent-to-purchase than those attached to either Verizon or Sprint. As mentioned earlier, the customer service "Geek Squad" table at the rear of the store was also predictive, indicating that employees were successfully able transition service visits into sales. I ran the product-category test from the previous section on a dataset with the Geek Squad dimension withheld to see if the Geek Squad help desk

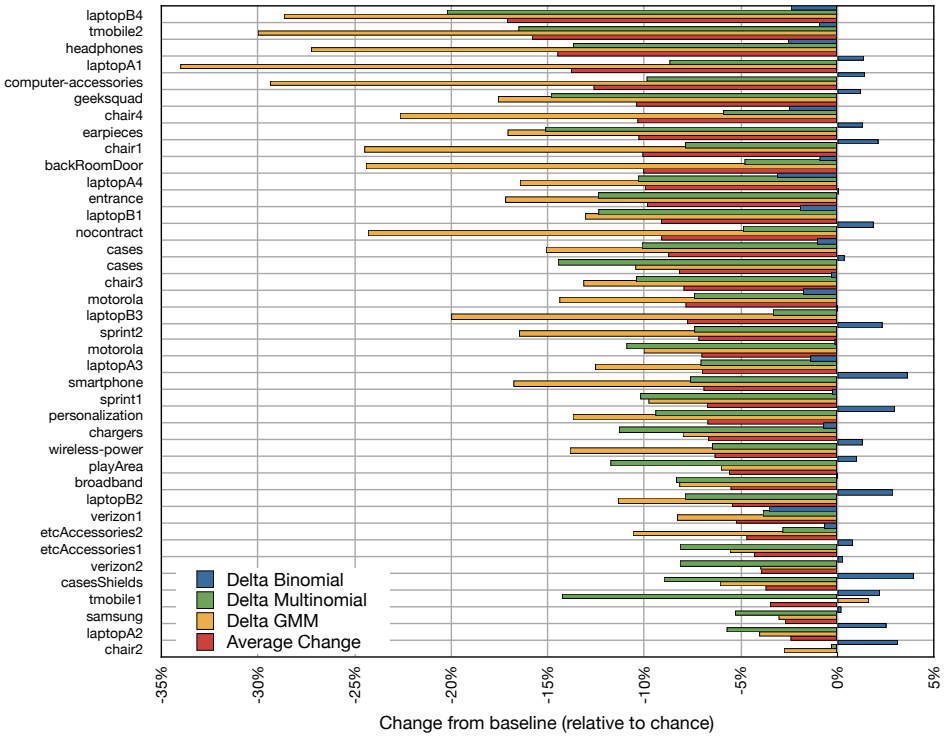


**Figure 5.12:** The relative importance of each functional location to the model's predictive power. This graph shows the change in model performance for each functional location when the floc was excluded from the input dataset. Positive values indicate an *increase* in performance without the floc. Negative values a *decrease*, indicating that the floc adds predictive power to the model. AUCs were calculated from the results of a twenty-fold cross-validation. Flocs are ranked by the average change in AUC across all three models.

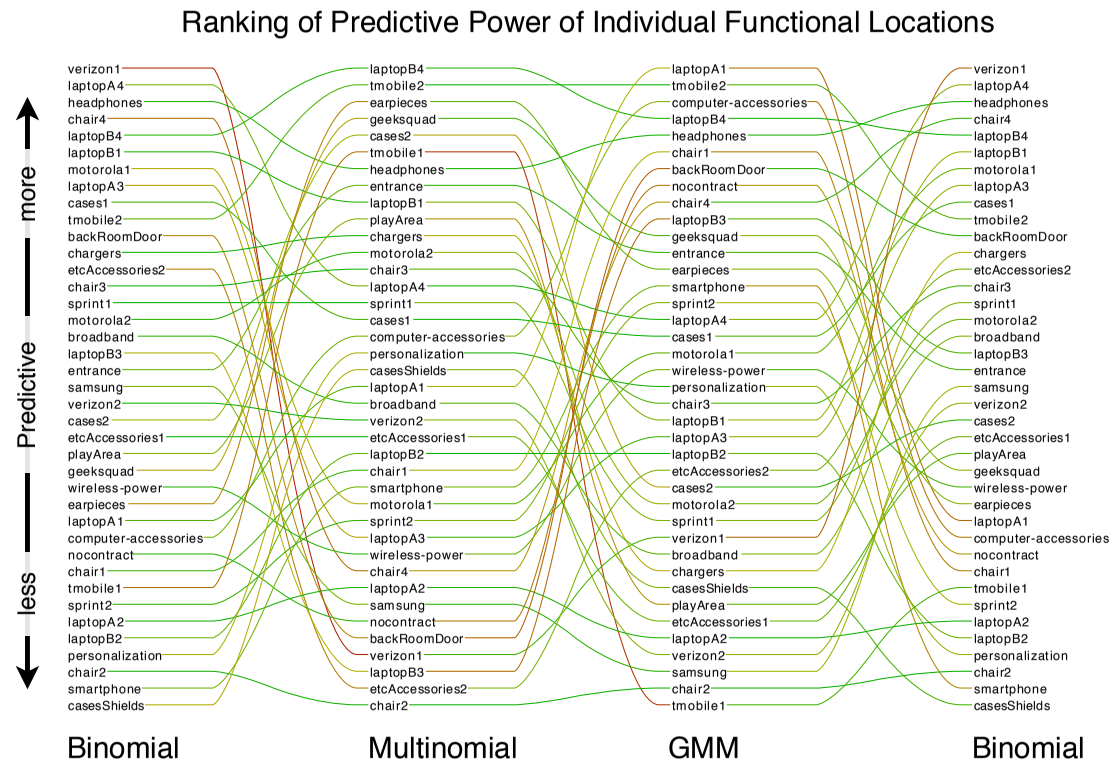


**Figure 5.13:** Ranking of flocs for the binomial, multinomial and Gaussian mixture based models. Colors indicate the degree of rank change. Green: little rank change; Red: significant rank change.

was particularly predictive of some SKU categories. The greatest negative differences in performance with and without the Geek Squad floc (indicating that the floc is informative for that category) came with Verizon hardware: a change from an AUC of 0.5919 to 0.5459—fifty percent different relative to chance. Next in order of percent change came T-Mobile contracts and headphones/MP3 speakers, with changes relative to chance of fifteen and fourteen percent, respectively.

Perhaps surprisingly, two of the four chair flocs appear in the top quartile of the ranking. At first scan, these flocs seem to have little to do with purchases. I believe the model picks up on several behaviors I observed personally on-site and in the recorded video. I noted many cases where a couple or pair of friends enter the store together where one has a particular need (e.g. find a fitting case for a mobile phone). The pair splits, one seeking the desired product, the other biding time reading a magazine while sitting in one of the chairs. Another common behavior occurred when the store was busy and all employees were occupied with customers. Then, patrons would sit in the chairs waiting for an employee to be available to assist the customer.

### 5.9.3 Layout Independence

How tied is the Mimic model to the particular configuration of the store? What is the connection between *function* and *location*? Fortunately, the Best Buy Mobile dataset offers the opportunity to test this question. During the recording of the Best Buy dataset, the physical layout of the store was changed significantly by moving a table of laptop computers to the front of the store. Unfortunately, there is an enormous confound which makes answering this question much more difficult: the holiday shopping season. The layout change first appears in the dataset on November 17, 2010, shortly before black Friday, the unofficial start of the holiday shopping season. As expected, the number of transactions following black friday and leading up to the Christmas holiday is higher than before (see Figure 5.14).

I performed a two-fold cross-validation of Mimic, where each fold contained episodes segregated by the physical layout of the store. This split divided the dataset into 700 positive and 562 negative examples of layout $A$, and 1056 positive and 951 negative examples of layout $B$ (See Figure 4.10 for an illustration of the two store configurations).

In the case of the binomial model, the change in layout had a small effect on classifier performance (AUC of 0.6164 with layout-based cross-validation vs. 0.6243 with randomized cross-validation), indicating that the model

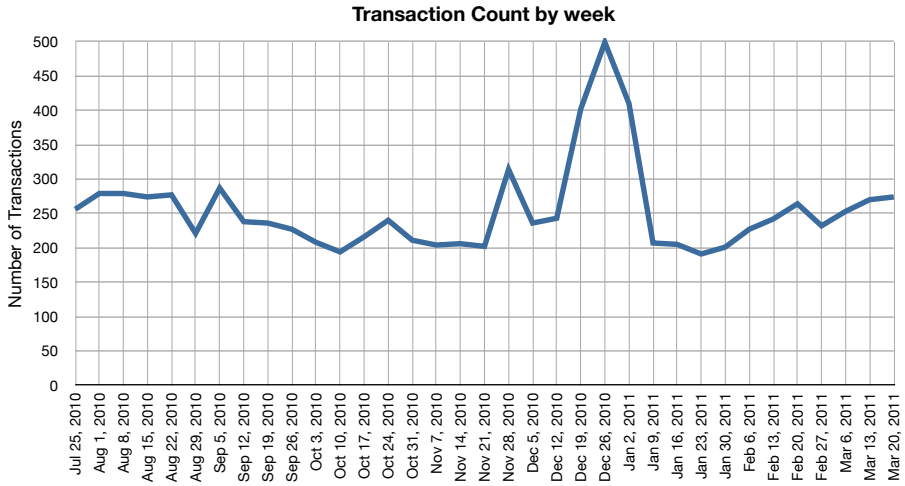


**Figure 5.14:** Weekly sales at the Best Buy Mobile Mall of America store. Of note are the two spikes in sales, one on the week of "black Friday" and the second preceding the Christmas holiday.

| Model Type | Randomized XV AUC | Layout-based XV AUC | (% change) |
|---|---|---|---|
| Binomial | 0.6243 | 0.6164 | -6.4% |
| Multinomial | 0.6070 | 0.5902 | -15.7% |
| GMM | 0.6158 | 0.5888 | -23.3% |

**Table 5.7:** Model sensitivity to store layout. Percent change is relative to chance.

captures layout-independent customer behavior. In contrast, both best-performing multinomial and GMM-based models suffered a more significant degradation in performance due to the change in layout (See table 5.7).

Models which do not suffer much performance degradation when the store layout is changed can be used as tools to introspect the sales-performance of a store (or several) as the layout is changed. The next chapter describes a tool which couples a simulator of customer behavior with the MIMIC model to create a tool that store designers and managers can use to optimize the physical layout of a store. This tool depends on a model whose performance is independent of the layout.

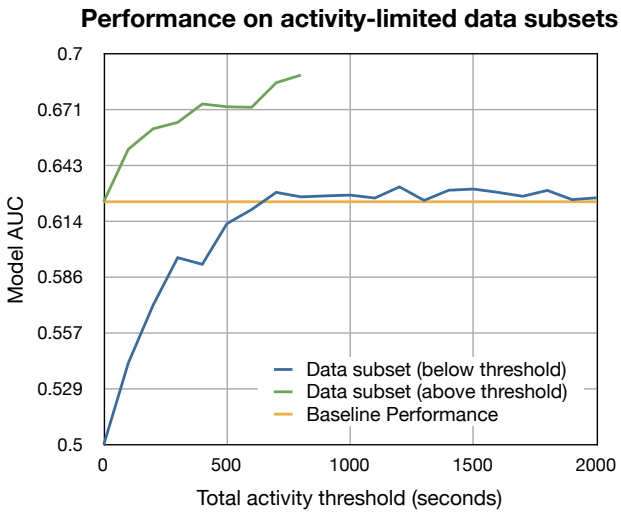


**Figure 5.15:** Performance of Mimic on subsets of the dataset. Episodes are included if their total activity is less than a threshold (which is varied here). Parameters of the best performing Mimic model (on the full dataset) were used.

### 5.9.4 Limiting Total Activity

Perhaps the Mimic model can perform better when limited to datasets where there are fewer people in the store. With fewer customers in the store, the floc activity data more closely resembles that of an individual, rather than the collective; it stands to reason that performance could thus be improved. To test this possibility, I cross-validated the best performing Mimic parameter settings on subsets of the dataset—the total activity serves as a proxy for the number of customers present in the store. An episode was included in the subset if its total activity (the sum of all floc features at all times) was less than a threshold. Data subsets were balanced as described in Section 5.4. I also tested subsets of the data where the criteria for inclusion was that the total activity was *above* threshold. Figure 5.15 shows the performance of the trained models as this threshold is varied.

With these parameter settings, the performance of the model trained the subset less than threshold did not significantly exceed baseline. With a low threshold, performance was significantly less, increasing to the asymptote of the baseline at a threshold of approximately seven-hundred total seconds. I believe that this trend can be explained by the high variance of the paths people choose as they walk around the store. When there are more people in the store, their aggregate activity pattern is closer to the true activity distribution (See Section 5.4). The model trained on the subset of data above threshold increased in performance as the threshold was increased (beyond eight-hundred total seconds, there were not enough examples to cross-validate

the model). This result supports the hypothesis that the variance of activity patterns strongly affects model performance.

### 5.9.5 Employee Proximity

In a store selling expensive and complicated goods such as the cell phones and mobile computers of the Best Buy case-study, employees have enormous impact. How much so? Can we improve the performance of Mimic by incorporating information about customer-employee interactions? I augmented the standard floc activity features with an additional "virtual" floc $v_{\text{interaction}}$ representing interactions between customers and employees. For every track labeled "customer" during a slice of time, I added the amount of time the track was within 1.5 meters of a track labeled "employee" to the virtual floc. As in section 3.2, for the time period $[t_1, t_2]$ and customer tracks $\{\tau_1, \ldots, \tau_m\}$:

$$v_{\text{interaction}} = \sum_{j=1}^{m} \{\text{duration } \tau_j \text{ was within 1.5 meters of an employee}\}.$$

Surprisingly, the inclusion of the employee proximity feature did not improve classification performance. On balanced data, the inclusion of the employee feature reduced the AUC of the best classifier from 0.6243 to 0.6187; on unbalanced data, the change was from 0.7400 to 0.7401. Comparing identical parameter settings on data with and without the additional feature, performance is almost always slightly less for data with the proximity feature. The same is true when run on SKU category subsets, as in Section 5.9.1: no classifier trained for a specific product category performs better than the same classifier trained without employee proximity information.

I propose two hypotheses for the lack of performance improvement. First, a lack of signal; employees converse with customers for a duration independent of an eventual purchase and with no significant difference in spatial distributions where these conversations happen between purchase and non-purchase conditions. An employee approaches entering customers for a greeting and offer of assistance irrespective of the customer's intent; an employee discussing the features of a phone or provider's service plans does so, again, whether a purchase is eventually made or not. Alternatively, the addition of this feature increases the model complexity such that more training data would be required to achieve commensurate performance. This second hypothesis is the so-called "curse of dimensionality." A preliminary attempt to incorporate employee-proximity features by splitting each floc feature in two, one for activity of customers alone, the second for activity

with customers near employees, also showed a decrease in classification performance.

### 5.9.6 Higher-level features: N-Grams

One possibly useful high-level feature extractable from trajectories alone are the common subsequences of flocs through which a person moves. The idea is to first identify the most frequent sequences of functional locations that customers pass through. The activity vector can then be augmented with additional features representing these subsequences; when a trajectory which contains a common subsequence passes into a floc, the corresponding dimension is increased as for a single floc alone. This is analogous to the $n$-gram language model from computational linguistics, where a sequence of $n$ words is assigned a probability. Here, a person's activity within a floc stands in for a word.

To establish the common subsequences, I transformed each track in the corpus into the sequence of flocs through which it passed (duplicated "words" in a floc sequence were removed), then tallied the count of all floc subsequences present. Unfortunately, less than 0.3% of all trajectories passed through the most frequent floc bigram, and less than 0.03% through the most common trigram. This dearth of significant common subsequences was likely due to two factors. First, the tracks in the corpus, though merged between multiple cameras, were often short. Second, the movement of customers within the store is hugely varied; the space of common subsequences is very sparse. For these reasons, floc $n$-grams are not very useful with this dataset. It would be worthwhile to reassess this conclusion when the quality of tracking and cross-camera merging is improved.

### 5.9.7 Finding Predictive Moments

What moment preceding a transaction is most helpful to making a classification? I used Mimic with a modified training set to find an answer. Rather than collect positive training examples from the $\psi$ seconds preceding a known transaction as in the standard model, I fixed the episode duration to five minutes, and varied how long before the transaction to collect the positive example episode. If there was a transaction at time $t$, then a positive episode was selected from the data between $t - (\psi + \gamma)$ and $t - \gamma$ (where $\psi$ was fixed at 300 seconds). Negative episodes were selected as for the standard model: activity data not used as positive data is divided into contiguous $\psi$-duration episodes and used as negative examples.

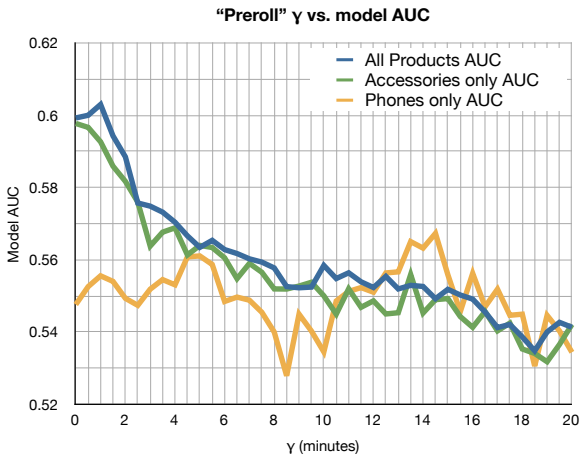


**Figure 5.16:** Predictive moments before a transaction. This graph was generated by evaluating MIMIC on five-minute-long positive episodes which preceded transactions by a variable amount. This graph shows that the moments immediately before transaction are most important for the MIMIC model.

The results shown in Figure 5.16 show that in general, the moments closest to a transaction are most informative. One potential explanation of this finding is that a large fraction of store sales consist entirely of cellphone accessories; a class of transactions which take far less time to complete than those that involve cell phone purchases and activations. To test this hypothesis, I ran two additional tests of predictive moments (also shown in Figure 5.16) using a positive datasets consisting only from transactions including mobile phone accessories, and only from transactions including mobile phone hardware from the three contract providers. Predictive moments for accessories-only match closely those for all transactions; in contrast, those for phones-only are more evenly distributed, with a slight performance increase centered approximately fifteen-minutes preceding transaction. This contrast supports the hypothesis.

## 5.10 Summary

The MIMIC model successfully extracts a weak signal from data filled with noise and confounds. Although the Hidden Markov Model of dynamic activity in the store gives great representational power, the performance of the learned models shows that simple models with few states most effectively capture

customer purchasing behavior. Even with a relatively weak signal, the Mimic model can be used to explore factors relevant to purchasing behavior such as the effect of layout, employees, and time. In the next chapter I elaborate on several other experiments and interactive tools which can use Mimic.

Chapter 6

# Conclusions

This thesis has reviewed consumer behavior models, introduced a new model, and demonstrated several results.

First, it introduced the retail world and the rich domain of pedestrian behavior analysis. With retail sales such a strong driver of the economy, even small changes to the efficiency of operations can have enormous impact on both on retailer (in terms of profits earned) and customers (in terms of dollars saved). The rapidly decreasing costs of data capture and storage, as well as the easy integration of contemporary computer vision and machine learning, enable new tools to better understand the behavior of customers—this thesis introduces one such tool.

Next, functional locations were introduced as a technique for capturing patterns of activity within a store in a compact format. The Mimic model, a discriminative classifier that can be used to predict customer transactions was presented. Finally, we saw an evaluation of Mimic on real data from a retail store.

Before delving into Mimic-enabled tools and recommendations for future work, let us revisit the central hypothesis of this thesis:

> The occurrence of a transaction can be predicted from the temporal *patterns* of activity distributions in the entire store. That is to say, there are measurable differences between the distribution of customers in a store preceding a sale and the distribution of customers when no sale occurs.

The previous chapter on the performance of the Mimic model validates this hypothesis. That Mimic is able to successfully extract signal from a stream

of data encoding patterns of activity demonstrates a measurable difference between purchase and non-purchase activity. The choice to examine whole-store patterns of movement rather than individual paths was motivated by the challenges of collecting accurate, continuous, pedestrian trajectories from overhead cameras. With raw video in uncontrolled settings, the current state of the art in tracking is accurate only for aggregate analysis such as that of Mimic.

In this chapter, I discuss three tools enabled by the Mimic model, and give a speculative proposal for future work that will best aid the efficacy of the model. Lastly, I summarize the contributions of this thesis.

## 6.1 Tools Enabled by Mimic

The performance of Mimic is far from perfect, but can still be used as part of a suite of tools that are immediately useful. Central to each of these tools is a philosophy of *human-computer collaborative tools*. In the context-sensitive, culturally-sensitive and hugely varied domain of human behavior—especially those domains dealing with video, as we have here—computational tools will likely long lag the near-instant insights available to even the untrained eye of the average adult. Computers though do not suffer from human limitations when coping with a deluge of data and herein lies the possibility for the human-machine collaboration: machines that aid humans by identifying (perhaps imperfectly) important snippets of data, by summarizing immense datasets, by using these libraries of past data to speculate about futures in a closed loop with a human partner. The three tools discussed below are examples of this style of collaboration, each for a different collaborative partner. The first, a smart engine for video retrieval, targets ethnographers and retailers who wish to hone in on consequential customer and employee behaviors that affect the bottom-line. The second tool helps managers compare and contrast the operations of multiple stores, providing a finer level of granularity than strictly transactional-based measures. Finally, the third tool targets store designers, helping optimize the physical layout of a store. Importantly, these tools are robust to imperfect results from Mimic.

### 6.1.1 Smart Engines for Video Retrieval

Retailers now collect hundreds of thousands of hours of video footage from their store every year, but ethnographers seeking to draw conclusions from this data are faced with the daunting task of sifting through this mountain of data to find interesting and relevant examples of behavior from which they

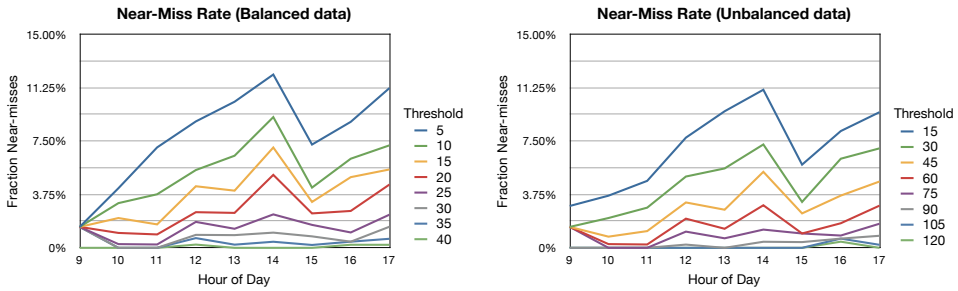


**Figure 6.1:** Transaction near-miss rate versus time of day. These graphs were generated using the best-performing MIMIC model for balanced and unbalanced data, respectively. The value at each hour represents the classifier false-positive rate for episodes during that hour for a given likelihood-ratio threshold $\gamma$.

can draw actionable conclusions. MIMIC can help winnow the overwhelming quantity of video to a more manageable size.

One way MIMIC can find interesting episodes is by identifying the near-misses: moments where the classifier had strong conviction that a transaction should have occurred, but for which no transactional record exists. These episodes correspond to false-positives toward the bottom-left corner of the classifier ROC curve. To give a sense of how such a tool could be used, I measured the false-positive rate for various likelihood-ratio thresholds ($\gamma$ in the model) as a function of time. I repeated this experiment using the best performing MIMIC model on both balanced and unbalanced datasets. The trend revealed in Figure 6.1 is that mid-afternoon and near closing are two times of day where Best Buy misses the most opportunities to make a sale. Perhaps these are times of day where the store could benefit from additional staffing.

What can be revealed by looking deeper than this aggregate? I examined raw video of the ten false positives episodes with the greatest log-likelihood ratio—those where MIMIC was most confident a transaction should have taken place. The first, second, and tenth episodes covered a contiguous thirty-minute block in early December. At that time, a special event was taking place—a promotional raffle for Virgin Mobile (a prepaid phone provider)—where over fifty people packed the front of the store. It comes as little surprise that no purchases were made during that time; the attention of both patrons and staff were on the proceedings of the event.

Several of the other highly-ranked false positives had a full staff of eight or nine employees and the store was relatively full. Often in those episodes, several employees would be occupied at the rear of the store assisting

customers, leaving fewer available to assist customers entering, browsing, or seeking specific products.[1] A specific suggestion for the retailer resulting from reviewing these videos is to streamline the process of phone activation or servicing, the execution of which currently takes in excess of twenty or thirty minutes. These long-duration interactions occur often during peak times. Recovering time by optimizing these procedures will free employees when staff is most in demand.

### 6.1.2 Realtime Tools for Managers

Managers responsible for several stores currently use basic metrics such as conversion-rate to evaluate the performance of their stores. Mimic may provide a more nuanced view, a systematic distillation of customer buying patterns beyond transactional evidence. For example, consider the near-misses identified by Mimic in the video-retrieval engine described above. A manager may use a store's near-miss rate to help identify problems in staffing schedules, training or inventory. Are there hours of the day with consistently higher near-miss rates? Are near-misses concentrated in certain product categories? Are those categories sufficiently in stock? Are there more near-misses when a certain subset of employees are staffed?

Mimic-based tools may be used to track the effectiveness of external or internal marketing campaigns: by how much did a Super-bowl ad drive traffic to a particular product display, and was there bleed in interest to other parts of the store? Did a new electronic display change the time spent interacting with a product or employee?

These types of tools can also help managers with a systemic understanding of stores under their purview. For example, staffing is one of the most significant costs in retail operations and is often handled by the local manager based on their intimate understanding of the store, its customers and their patterns. But are there systemic problems present across many stores for which a higher-level manager can give guidance? Are there common times of day where the employee to customer ratio falls below a desired threshold? Which store managers are able to best anticipate the staffing needs; identified, what best practices can be learned from them? Which store managers are worst performers and could benefit from additional training?

[1] Without audio capturing conversations with customers, it is difficult to judge specific customer intent or the reasons behind an interaction between employee and customer.

### 6.1.3 Store Layout Optimization

A store designer's role is to strike a balance among the many factors that shape a store's layout. There are aesthetic considerations, contractual ones (e.g. leased spaces in the store like aisle end-caps), functional and narrative. This last consideration being the stories of a customer's journey in the store.[2] But the ultimate measure for a retail space is the bottom line. The complex contribution of all these factors to the bottom line may forever be out of the reach of computational models, though Mimic-based tools can augment the intuition and experience of designers with a data-driven gauges of gross performance.

The idea is to couple Mimic with a simulation of customer behavior. For example, consider an agent-based simulation of customer behavior like that shown in Figure 6.2 where a simulated human navigates a store to a preset goal chosen based on the transactional record.[3] The activity patterns of hundreds of these virtual customers can be evaluated by Mimic to provide a "score" for the store layout used in simulation. Alternatively, a discretized stochastic simulation such as the model used in (Farley and Ring, 1966) can generate analogous activity-patterns for Mimic to score.

Agent-based simulations are currently used to evaluate building performance along narrow dimensions such as traffic flow (Smith et al., 1995) and safety (Helbing et al., 2000) or for communicative purposes (Burkhard et al., 2008; Narahara, 2007), but only very recently to capture higher-level and more complex behavior such as shopping. Terano et al. developed such an agent-based model for a supermarket; in their model, purchase decisions were baked into the agent design rather than being driven by a coupled activity-transaction model like Mimic (Terano et al., 2009).

One could imagine closing the loop entirely by integrating an automatic layout engine whose output is iteratively improved by using Mimic as a fitness function. Layout algorithms go back at least as far as Eastman's GSP (Eastman, 1971) and the shape grammars of Stiny and Mitchell (1978), and have involved classic artificial intelligence techniques such as expert systems (Gullichsen and Chang, 1985) and genetic algorithms (Jo and Gero, 1998).

[2] Narrative also refers to the experiential. The showroom floor of an IKEA furniture store is a good example—the zig-zagging path a customer travels through the showroom is a very explicit expedition from one imagined lifestyle to next.

[3] The name "Mimic" is the legacy of an agent-based model I developed earlier in this research trajectory. Some good introductions to agent-based modeling and their use in microscopic pedestrian analysis and simulation can be found in Bonabeau (2002) and Kerridge et al. (2001).

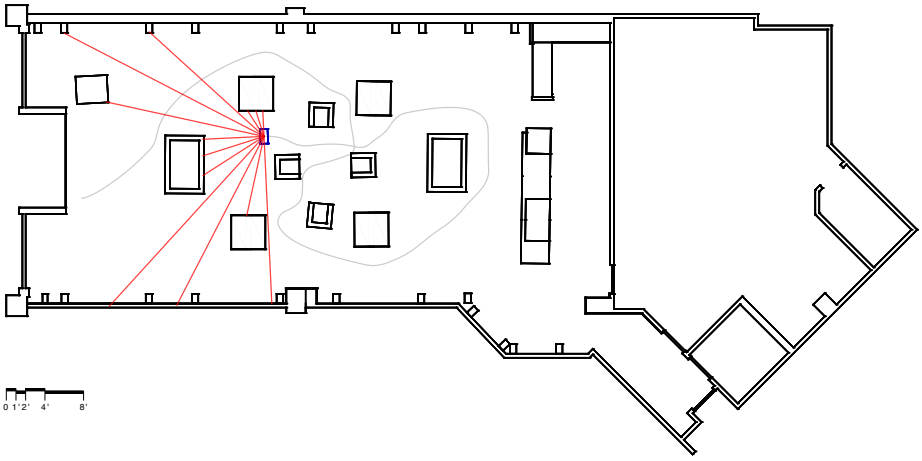

**Figure 6.2:** Agent-based simulation may be coupled with the MIMIC model to optimize the sales performance of the store. In this image, a simulated customer (the small blue rectangle) has entered the store and is navigating to several pre-programmed destinations. This agent uses a variant of the social-force model of (Helbing and Molnár, 1995); the red rays emanating from the agent represent its field of vision.

Even in a more automated layout optimization scheme, a human designer must always be in the loop to critique from the uniquely human perspective. Such a simulation/evaluation framework is an example of the human-machine collaborative effort.

## 6.2 Future Directions

The problem-space engaged in this thesis is enormously rich, and rife with possibilities for future directions. The experience exploring the parameters of the MIMIC model, and examining in some detail the failure modes of the classifier, may give insight into which directions may prove the most fruitful for future researchers.

### 6.2.1 Three Suggestions for High-value Improvements

Based on the experience working with the Best Buy case study dataset, I believe the following three suggestions for improvement of modules in the MIMIC pipeline would have the greatest impact on the system's capabilities.

The first focuses on low-level features input to the system, the second on data filtering, and the third on high-level temporal modeling.

**Tracking Improvements:** The choice to examine the overall activity patterns in the store rather than individual customers was motivated by the challenges of tracking people in video and accurately stitching their tracklets into complete paths within the store. Accurate, complete trajectories would have a transformative effect on Mimic's predictive power. For example, positive training episodes examples could be accurately extracted using the actual duration of a purchasing-customer's store visit, rather that the coarse-grained fixed value used thus far.

More importantly, rather than model the pattern of aggregate activity, we could model the paths of individual trajectories. This would give much more focused insight into what factors affect the purchase decision; we would be able to bring a scalpel rather than Mimic's hammer. We would be able to answer questions such as:

- Does precedence affect a purchase decision? I.e. when comparing two competitor products, does it matter which is examined first? Last?
- What kinds of profiles of engagement between customer and employee are best? Many stores employ a greeter, who welcomes customers into the store with a simple "hello." Does the presence of a greeter have any effect, positive or negative on a customer's propensity for buying?
- What are the differences in purchasing behavior of customers arriving in groups as opposed to those arriving alone?

Some studies, such as Larson et al. (2005) and Hui et al. (2009a), performed in supermarkets with RFID tracking technology which provides continuous and complete trajectories, have begun to answer these sorts of questions, but video is an immensely more rich medium and more applicable in some retail situations. This leads to the second suggestion.

**Pedestrian Detection & Classification:** Currently, Mimic considers pedestrians in the store as either customer or employee, with no finer-grained division. From video, we can now computationally determine gender (Mäkinen and Raisamo, 2008), race (Shakhnarovich et al., 2002), age (Fu et al., 2010; Kwon and Lobo, 1999), affect (Fasel and Luettin, 2003) and even gross body language. These tools can be added as filters to the model underlying Mimic,

and with the massive video corpora being captured in retail today, one could expect sufficient data to draw actionable conclusions.

Subcategorizing customers in the store—inferring age, gender, ethnicity, etc.—would impact both explanatory and predictive power; explanatory in that researchers inspecting the a model's workings could gain quantitative understanding of behavior across demographics, and predictive in that these unmasked differences could lead to improved Mimic classifications.

The overhead, omnidirectional, low-resolution views provided by the cameras of the case study store make demographic classifications challenging, though additional cameras at eye-level could be incorporated without much difficulty (classifications made using these cameras could be propagated to tracks traveling outside the camera's range).

**Alternative Temporal Models:** The particular formulation of Mimic as an HMM is but one of many possibilities. Other modeling choices may be more performative. For example, rather that a pair of HMMs, one for positive and negative examples, a single HMM could be trained on the combined data, where one of the states represents *transaction*. During training, the ultimate HMM state would be forced to this *transaction*-state for the positive examples. An episode could then be classified by finding the most likely sequence of HMM states, then choosing a label based on the final state.

One of the drawbacks of Hidden Markov Models is their difficulty capturing long-distance dependencies between observations—dependencies which very likely exist in a store dataset. A trivial example from the Best Buy dataset: a customer shopping for a new phone who visits two different service-provider's kiosks is likely to visit the third provider's as well; this dependency exists at a distance of several minutes.

A mixture of HMMs may disentangle the behaviors of individuals from the summation of many stands of individual activity that make up the activity-pattern data-stream. In such a model, training data of individual customers movements are used to train a collection of HMMs which become the mixture components.

Finally, the classification task can be framed as a discriminative rather than generative problem. Other temporal modeling techniques such as conditional random fields may better capture the behavior patterns in a store (Lafferty et al., 2001).

### 6.2.2 Experiments Using MIMIC

Chapter 5 included several examples of explorations using the MIMIC model. These investigations are only the tip of a much larger suite of experiments. The following experiments may be of interest to both retailers and ethnographers.

- A important stream of revenue for retailers is the renting of specific locations within the store to the companies whose products are being sold. An example are the *end-caps* at the end of aisles—due to the greater visibility they provide, these locations see greater sales and demand a higher rental price. MIMIC can quantitatively answer the question: What is the value of these spaces within the store, not only for the specific product located there, but to sales throughout the store. This experiment is similar to the one in Section 5.9.2 which examined the sensitivity of the model to individual flocs, but more narrowly focused to a subset of products.

- Similarly, MIMIC can show the influence of unrelated flocs on sales. An example that would be relevant to the Best Buy Mobile store: say a particular brand of Bluetooth earpiece is compatible with a wide variety of cellphones. This product is physically located in one place in the store—an area encompassed by one functional location. A MIMIC model can be trained with positive examples derived only from those transactions which include the product, but using activity pattern feature vectors that *exclude* the product's floc. An examination of the learned model will reveal which other parts of the store may influence purchase of the earpiece. This guidance helps both customer and retailer in that complementary products can be suggested to the customer. Product affinities may be more visible than in the transactional record.

- In Section 5.9.1, MIMIC was trained with data of transactions containing specific product categories, revealing potentially valuable insights about customer behavior and its relation to the physical environment. This analysis can be automated to discover other outlier behavior patterns and be used in situ, giving feedback to store managers and designers. Specifically, a marked difference in performance between a baseline (all-transaction) and category-based model can be flagged for review. Likewise, significant differences between static distributions coded by the baseline and category-based HMM states may also be worthy of further analysis.

### 6.2.3 Other Stores: Several Conjectures

The case-study Best Buy Mobile stand-alone store represents just one corner of the landscape of retail stores. The store is physically small, with a large assortment of products that are expensive and infrequently purchased. In that sense, the Best Buy Mobile store resembles a jewelry store. How might we expect the MIMIC model to perform in other retail environments, for example, in a big-box retailer such as Walmart, or the supermarket Shaws?

A larger store will have many more functional locations, and the increase in dimensionality of the activity vector will likely obfuscate discernible patterns. From the perspective of a customer in a large store, this makes sense: activity local to, say, the men's shoes section is invisible to (and has no real influence on) someone a hundred meters away in the power-tools section. Another challenge posed by larger stores is the much longer and more variable time between when a customer handles a product and when the purchase is made and recorded in the electronic record.

Both of these challenges point to the need for a model which uses full-length trajectories of individuals, rather than the aggregated activity of the store as a whole.

## 6.3 Contributions

The three major contributions of this thesis are:

- The introduction of *functional locations*, and the measurement of activity taking place within them, as a low-dimensional feature which effectively captures meaningful patterns of activity within a store.

- MIMIC is an example end-to-end prediction system, tested on real-world data, that is able to foresee the occurrence of transactions from video and transactional information captured in a store. Individual modules within the system can be replaced with higher-performing components with the expectation that the overall performance of the classifier will improve.

- The end-to-end transaction prediction model presented in here validates the hypothesis: patterns of activity of *an entire store* are indicative of customer intent-to-purchase.

## 6.4 Final Words

Mimic is an immediately applicable tool. Retailers already have an extensive infrastructure of video cameras in their stores; integrating a system like Mimic into their existing infrastructure is an incremental addition. Doing so would arm retailers to increase their sales, reduce costs, and make a better experience for their customers. Tools such as Mimic can have significant financial impact. For a large retailer such as Best Buy, a two percent change in store conversion rate translates into a billion dollar shift in revenue.[4]

The techniques discussed here—the use of functional locations, patterns of activity and their integration with timestamped electronic records—also have the potential to influence the design and operation of places outside retail. Imagine a museum delivering to your cellphone tour recommendations customized to your interests based on your dwell patterns... Imagine an airport made easier to navigate via the simulation of thousands of travelers like you... Imagine a hospital which helps prepare itself for emergency triage after recognizing the behaviors of doctors and nurses after a major accident is called in...

[4] Neil McPhail, Sr. VP and general manager at Best Buy, personal communication.

Appendix A

# Data capture

Joint work with George Shaw and Philip DeCamp.

The following are some technical details about the data capture pipeline not included in earlier chapters.

## A.1 The Data Pipeline

A more detailed diagram of the data flow is shown in Figure A.1. A brief description of each components follows:

**Camera** Lumenera IP web-cameras captured video at 960 by 960 pixels. Images were compressed by the camera using JPEG.

**Recording Servers** Each recording server serviced two cameras, pulling frames from the cameras and packaging them into minute-long files stored locally.

**Transport** Periodically, data-filled hard drives were shipped from the Best Buy store to MIT.

**Video Repository** Video files were stored on a shared file server at the MIT Media Lab.

**Video Proxy Converter** Generation of low-resolution proxies (used in tracking) was primarily the responsibility of the recording servers. Proxies were sometimes removed to free space on the local recorder

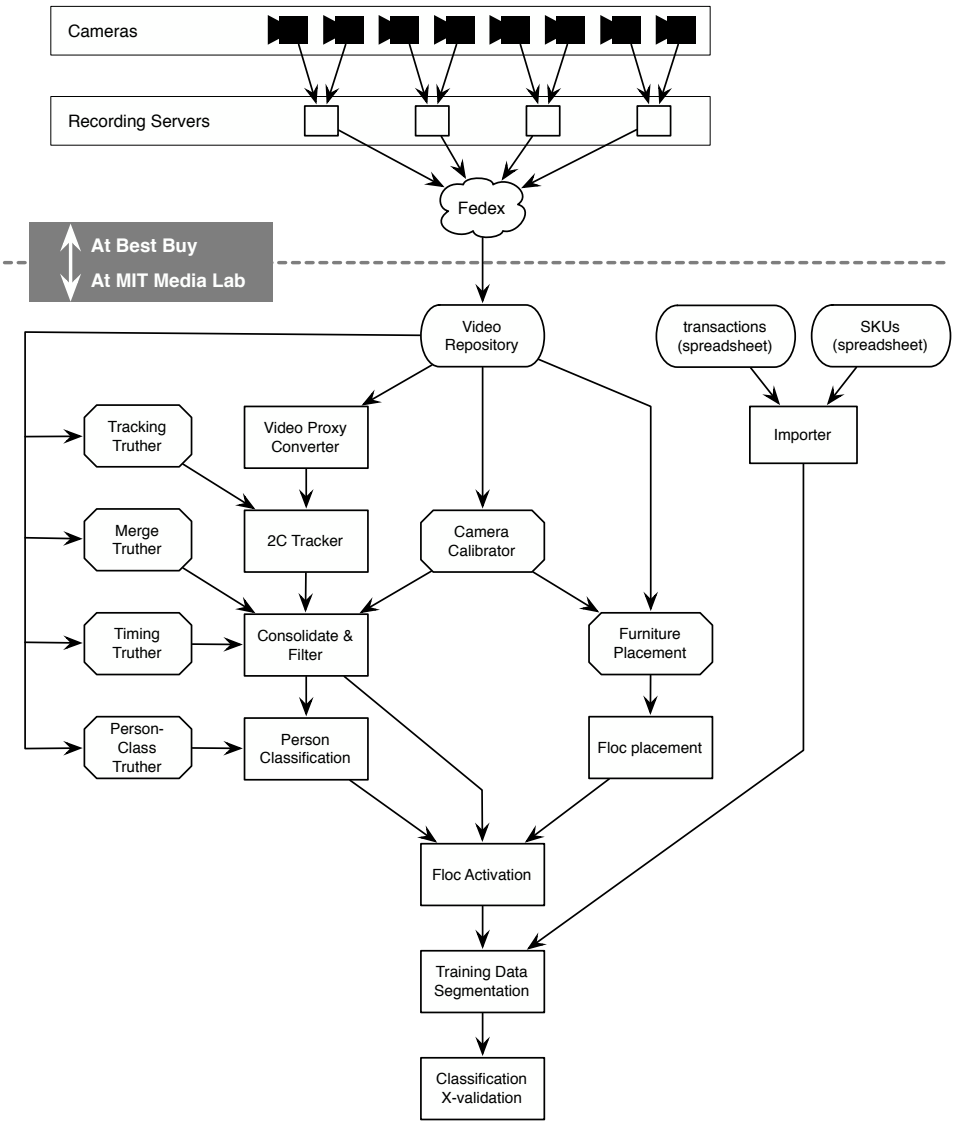


**Figure A.1:** Detailed data-flow pipeline

drives and needed to be regenerated at MIT. This automated process recreated low-resolution video proxy files as necessary.

**Tracking Truther** This annotation tool allows person trajectories to be manually created. Trajectories so generated were used as a ground-truth to tune the parameters of the 2c tracker.

**2c Tracker** The tracking module which generated person-tracks for each of the eight cameras. Tracking was completed on a heterogeneous cluster of server and desktop machines. The entire corpus could thus be re-tracked in a matter of hours.

**Merge Truther** An annotation tool to collect a ground-truth training set for camera tracks that target the same person and so should be merged. The tool works by first generating an expansive set of candidate trajectory pairs which are presented to the user, who accepts or rejects each pair in turn.

**Camera Calibrator** A tool to associate points in an image with coordinates in the global Euclidean frame. A screenshot of this tool can be seen in Figure 4.6 on page 59. The camera calibrator optimizes parameters of the camera location, orientation and lens characteristics.

**Person-Class Truther** An annotation tool to collect training data for the customer/employee classifier. Tracks are shown one by one superimposed on video. The user selects whether the track targets a customer or employee.

**Consolidate & Filter** This module comprises several steps of post-processing of the tracks generated by 2c. Tracks were first filtered to mitigate commonly-seen tracking errors. Very short tracks were removed as were track points intersecting manually-drawn masks. These masks cover televisions and computer displays in the store, locations physically impossible to access and transient error sources such as helium balloons in the field of view. Tracks were next smoothed with a Kalman filter and merged using the scheme described in the following appendix.

**Furniture Placement** An annotation tool used to place movable furniture and fixtures in the store. Every moveable fixture onto which a floc was attached was manually placed for each day in the recorded corpus.

**Person Classification** The automatic classifier which separated customers and employee tracks.

**Floc Placement** This module automatically aligned functional locations with the annotations of furniture locations.

**Floc Activation** Here, customer trajectories were processed into a stream of floc activation features (Section 3.2).

**Transactions & SKU Importer** An importer of transaction data and the taxonomy of products from Best Buy. Data from Best Buy came as spreadsheets; this importer converted these into a SQLite relational database file.

**Training Data Segmentation** The output of the Floc Activation module is a sequence of activity feature vectors for an entire day. This module divided the feature vectors into positive and negative example episodes of fixed length given the transactional data. The distribution of total activity was balanced between positive and negative training episodes in this module.

**Classification Cross-validation** This final module evaluated the HMM-based Mimic model using the training episodes provided from upstream. Here, test-set episodes were excluded if they temporally overlap with training data.

## A.2 Hardware Setup

Video was captured in the Best Buy Mobile store using Lumenera Le165C cameras and custom recording software written by Philip DeCamp. Each camera compressed the 960 by 960 image onboard using JPEG and appended a timestamp. Images were fetched from the eight cameras via HTTP by one of four Apple Mac Mini servers. These servers then packaged the JPEGs into files, one for each minute and for each camera. The servers also generated low-resolution proxy images at 120 by 120 pixel resolution. Video was recorded to external hard drives which were shipped to MIT when filled. A fifth Apple mini served as a central controller and NTP server for the four recording machines.

## A.3 Camera Synchronizations

Each camera in the BestBuy Mobile store was synchronized to an NTP server running on the local control server. Unfortunately, during the course

of recording, this server failed, and the cameras slowly drifted out of sync before the problem was noticed and corrected. Synchronization of the cameras to each other is critical to downstream components of the Mimic pipeline, especially the handing off of customer trajectories from one camera to the next.

I built an annotation tool to manually to resynchronize the cameras. The tool allows an annotator to find pairs of frames where two cameras recorded the same moment in time—this was made possible because each camera's field of view overlaps those of adjacent cameras. The annotator can quickly mark a simultaneous footfall or other synchronous event. The output of the annotation tool is a collection of tuples each of which consists of the IDs of two cameras and a difference in their time-bases; essentially, a graph with nodes representing cameras, and directed edges denoting annotations, with edge weights being the time-base difference. To re-synchronize the data from all eight cameras to the same clock, I devised a simple weighted-averaging algorithm to determine a single offset for each camera.

The nodes of the annotation graph $G$, representing the $n$ cameras are $\{c_1, \ldots, c_n\}$. An edge $e_{ij}$ is an annotation, whose's weight $w_{ij}$ is the time offset between cameras $i$ and $j$. The algorithm proceeds as follows:

**Algorithm 2** Procedure for calculation camera timestamp offsets.

Ensure that for every directed edge $e_{ij}$ with weight $w_{ij}$ in the graph, there exists an edge $e_{ji}$ with weight $-w_{ij}$.
Choose one camera, $c_0$ to be the anchor—to have zero time offset.
**for all** each other camera node $c_i$ **do**
  $S_i \leftarrow 0, T_i \leftarrow 0$
  **for all** distinct non-repeating paths $p_i$ through the graph $G$ from $c_0$ to $c_i$. **do**
    $\delta t \leftarrow$ the sum of weights of the $p_i$'s edges.
    $\omega \leftarrow |p_i|$ {The number of edges in the path.}
    $T_i \leftarrow T_i + \frac{1}{\omega}\delta t$
    $S_i \leftarrow S_i + \frac{1}{\omega}$
  **end for**
  The time offset for $c_i \leftarrow \frac{T_i}{S_i}$
**end for**

Camera time offsets were then applied to tracks during post-processing (see below).

## A.4 Tracking

The tracking module, dubbed 2c, used in this thesis was developed by George Shaw as part of his master's thesis and is fully described in (Shaw, 2011, pages 34–53). Tracking proceeds through the following simplified pipeline for each frame of video:

**Input:** Rather than use the full resolution (960 by 960 pixel) images, tracking was performed on a lower resolution proxy (120 by 120 pixels). This significantly lowered the the memory requirements and increased tracking speed. Low resolution proxies were generated at recording-time.

**Background/Foreground classification:** I replaced the original mixture of Gaussians background model with an implementation of the foreground-adaptive probabilistic background detection described in (McHugh et al., 2009). A cyclic buffer of past frames forms the basis of the background model. The buffer holds forty frames, and is updated every 400 frames. An initial probability of background is calculated for each pixel by comparing it to the corresponding pixel in the buffered frames. The foreground model is initialized by thresholding this initial probability, and then further refined by iteratively examining a small neighborhood around the candidate (5 by 5 pixels), increasing the foreground probability if there are nearby foreground-labeled pixels. A final pass of a Markov random field smooths the resulting foreground/background classification.

**Particle Generation:** Pixels labeled *foreground* are clustered into *particles*. For every small square patch of pixels, a particle is created if the ratio of foreground pixels exceeds a threshold. The particles are then clustered using connected components.

**Particle Association:** Particles from the current frame are associated with tracklets which were incrementally built from previous frames. This association step incorporates location, color, shape and motion features.

**Export:** Tracks and color histograms are exported to SQLite database files. This component ends the 2c tracking pipeline.

## A.5 Track Post-processing

Tracks generated by 2c were processed by the following filters.

**Point Masking:** Track points intersecting manually annotated image masks were removed. Masked areas included inaccessible parts of the image (e.g. ceiling), video monitors and other false-positive noise sources.

**Minimum Duration:** Tracks were required to have a minimum duration of one second.

**Minimum Point Count:** Tracks of less then ten points were discarded.

**Motionless Filter:** Any track points that remained motionless for longer than seven seconds were removed (to reduce false positives due to patrons becoming incorporated into the background model).

**Maximum Gap:** After the motionless filter is applied, there may exist tracks with significant temporal gaps between points. This filter cuts these tracks whenever a gap exceeds ten seconds, discarding all points after the gap.

**Coordinate Transform:** Tracks are translated between image coordinates and a global Euclidean frame. Image coordinates are preserved for any downstream processes.

**Kalman Filter:** A filter that spatially smooths the trajectories in the global coordinate frame. I made the simplifying assumption that tracked motion exists in the plane parallel to the floor. The Kalman filter's measurement noise represents a cone of confusion of constant angle extending from the camera in the direction of the target. The intersection of this cone with a plane at the height of the target (fixed to one meter) determines the covariance of the measurement noise.

**Time synchronization:** Temporal offsets were applied to tracks to synchronize them to a common clock. These offsets are calculated using annotated data from the Time-Truther module (Section A.3).

**Customer/Employee Classification:** A track's accumulated color histograms were used to classify the track target as either customer or employee. The results are stored in a table in the output SQLite file.

Post processed tracks, like the raw output of 2c, are stored in SQLite files.

## A.6 Projection to a Global Coordinate Frame

A spherical camera model was used to calibrate the surveillance cameras used during recording. This model better fits the distortion of the fisheye lenses used in recording than a traditional pinhole homography model. Details on the camera model, and the calibration procedure can be found in DeCamp et al. (2010).

To calibrate the eight cameras installed in the Best Buy, I built a 3D CAD model of the store from 2D plans provided by Best Buy (Figure 4.6 on page 59), and detailed measurements taken on site. Cameras were then calibrated by finding intrinsic (lens) and extrinsic (location and orientation) camera parameters that minimized the reprojection error of a collection of annotated correspondence points between camera $\{x, y\}$ image coordinates and Euclidean $\{x, y, z\}$ points (read from the 3D model). These parameters were found using the Levenberg-Marquardt solver provided in the MINPACK library (Moré et al., 1980).

Tracks from a single camera only give two dimensions (image-$x$ and image-$y$) which can be used to define a ray extending from the camera. To locate the target of the trajectory in three dimensions, a multi-scopic estimate was made for those trajectories comprised of a merging of several tracklets from different cameras (see Section 4.3). An estimate of the location was made for those trajectories that were not merged across cameras by intersecting the ray from the camera in the direction of the target with a plane at a height of one meter.

Appendix B

# Tracklet Merging

Joint work with Matthew Miller.

As a person walks in the store, she is captured on video simultaneously by several cameras. The 2c tracker outputs tracks for her from each of the cameras. All these tracks need to be consolidated into a single unified trajectory in a global coordinate system. When there is more than a single person present, this presents the problem: which tracks should be merged together, and which represent different people? This is the track-merging problem.

We use the nomenclature of *tracklets* and *tracks* to make the distinction between trajectories captured from a single camera, and those agglomerated trajectories from multiple cameras, respectively.

We framed the cross-camera merging problem as a classification task: Given two tracklets, determine whether or not they should be merged together. In the following descriptions, the tracks are denoted $\tau_1(t)$ and $\tau_2(t)$, where $\tau(t)$ is the location vector $\{x, y, z\}$ at time $t$. Two tracklets are candidates to be merged if:

- They were generated from different cameras.
- They overlap temporally.
- The $\{x, y\}$ bounding box enclosing each tracklet overlap.
- They pass a small set of manual criteria. For example, the average distance between the two tracklets must not exceed a large threshold.

These criteria bound the number of tracklet-pair candidates to be approximately linear with the total number of tracks.

A feature vector encoding the pairwise relationship between two tracklets at a given moment of time is $f_t(\tau_1(t), \tau_1(t))$. For every tracklet pair, we calculate $f(\cdot)$ at fixed time intervals between $t_0$ and $t_1$, the times where the tracklet pair overlaps temporally. The vector $f$ has the following dimensions:

**Distance** The Euclidean distance between the two tracklets: $|\tau_1(t) - \tau_2(t)|$

**Distance-squared** The square of the distance dimension. In the Gaussian mixture model, this dimension penalizes those track pairs with large delta-distance.

**Log-Distance** The log of the distance dimension. In the Gaussian mixture model, this dimension penalizes those track pairs with small delta-distance.

**Square-root Max delta distance** Equal to the square root of the mean of the top 5% distance features between the two tracks. The idea here to capture those cases where a pair of people walk together for a time, then diverge; the top delta-distance captures the portion of the tracks which diverge. The root better clusters high delta-distance values.

**Optimistic delta-distance** Similar to the distance metric, but instead uses the shortest distance between the two lines defined by the location of the camera and the image patch being tracked.

**Delta-velocity** The magnitude of difference between velocities.

$$\text{D.V.}(t) = |\dot{\tau}_1(t) - \dot{\tau}_2(t)|$$

**Velocity dot product** $\text{V.D.P.}(t) = \dot{\tau}_1(t) \cdot \dot{\tau}_2(t)$

**Delta Shape** Calculated similarly to the optimistic delta-distance, where first the two tracks are aligned by subtracting the mean delta-distance.

**Delta angular velocity** The magnitude of the difference of angular velocities of the two tracks.

**Overlap time** $t_1 - t_0$. This value is constant for all times between $t_0$ and $t_1$.

**Angular velocity product** The product of angular velocities of the two tracks.

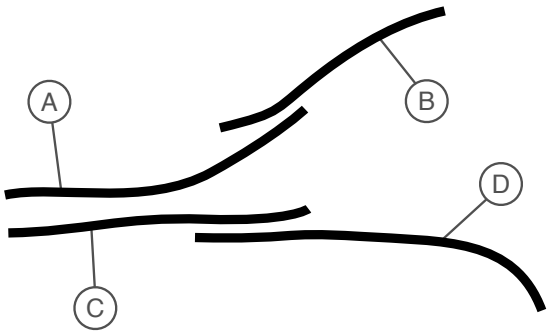


**Figure B.1:** A hypothetical tracklet-merging example which illustrates the challenges of finding a global solution from local decisions. Here, track pairs $A$–$B$, $C$–$D$, and $A$–$C$ would be classified to be merged when using a myopic merger.

**Delta Histogram** The KL divergence between the aggregate color histograms of the two tracks. This value is constant for each time between $t_0$ and $t_1$.

We used a Gaussian Mixture Model (GMM), trained via expectation-maximization, to model the likelihood that a tracklet pair belongs to either the *merge* or *no-merge* classes. To train the model, I manually annotated a set of 1928 tracklet pairs that should be merged. All tracklet pairs that passed the inclusion criteria described above and were not in the "to-merge" annotated set were considered negative examples for the training set. The GMM was parameterized with one hundred mixtures.

The Gaussian mixture model classifier is the first component of the merging algorithm. Next, we need a mechanism that can combine these local decisions into a global solution. Figure B.1 shows an example of four hypothetical tracklets. These tracklets may have been generated by two people who walk into the space together, then separate. An optimal solution would merge tracklets $A$ and $B$, as well as $C$ and $D$. But, from this illustration, tracklets $A$ & $C$ are attractive candidates for merging. The next component in the merging algorithm solves this problem.

More generally, if we have two track clusters $\mathbf{T}_1$ and $\mathbf{T}_2$, each consisting of a bundle of tracklets, and we have one tracklet pair $A$–$B$, where $A \in \mathbf{T}_1$ and $B \in \mathbf{T}_2$, how can we use a myopic *tracklet*-merging classifier to make a macroscopic *track*-merging classifier?

We calculate the likelihood of merge and non-merge for the two clusters $\mathbf{T}_1$ and $\mathbf{T}_2$ as the product of likelihoods of the local decisions. Suppose the parameterization of a class' GMM is $\theta$, and the tracklet members of the

clusters are $\{\tau_{11}, \ldots, \tau_{1m}\}$ and $\{\tau_{21}, \ldots, \tau_{2n}\}$ respectively. The likelihood of the class is then

$$P(\mathbf{T}_1, \mathbf{T}_2|\theta) = \prod_{i=1}^{m} \prod_{j=1}^{n} P(\tau_{1i}, \tau_{2j}|\theta)$$

This calculation is repeated for the merge and no-merge models.

Our solution to solving the global merge problem is an iterative greedy merger. The greedy merging algorithm proceeds as follows:

**Algorithm 3**

Initialize track clusters: $\forall \tau_i, \mathbf{T}_i = \{\tau_i\}$
Find all tracklet pairs that pass the initial merge criteria.
**repeat**
  **for all** remaining merge-pair $\{\tau_i, \tau_j\}$ **do**
    $\mathbf{T}_i \leftarrow$ the track cluster containing $\tau_i$
    $\mathbf{T}_j \leftarrow$ the track cluster containing $\tau_j$
    Calculate the likelihood of classes *merge* and *no-merge*:
    $p_{\text{merge}} \leftarrow P(\mathbf{T}_i, \mathbf{T}_j|\theta_{\text{merge}})$
    $p_{\text{no-merge}} \leftarrow P(\mathbf{T}_i, \mathbf{T}_j|\theta_{\text{no-merge}})$
  **end for**
  $\{\tau_i, \tau_j\} \leftarrow$ the pair which maximizes $p_{\text{merge}} - p_{\text{no-merge}}$
  **if** $p_{\text{merge}} - p_{\text{no-merge}} > 0$ **then**
    Merge clusters $\mathbf{T}_i$ and $\mathbf{T}_j$ which contain $\tau_i$ and $\tau_j$.
  **end if**
**until** No pair is classified as class *merge*: $p_{\text{merge}} - p_{\text{no-merge}} \leq 0$.

On a test set, the greedy merge classifier performed with 86.2% accuracy (see Table B.1), precision 71.0%, and recall 38.3%.

The high level of accuracy achieved by the merger is somewhat deceptive. As a person walks through the store, he may generate tens of tracks, being visible from many cameras. The probability of an error in the merging process is quite high,[1] making nearly impossible to have accurate continuous trajectories of people in the store. The features used in this merging algorithm are very coarse-grained and based almost entirely on the characteristics of the trajectories themselves. More sophisticated features, such as those that could be derived from facial features, gender, age, or ethnic classifiers, could

[1] To make this error-rate concrete, imagine a customer who enters the store, walks along the perimeter of the store, then exits. There will be at least eight hand-offs between cameras. With the accuracy garnered by our tracklet merging algorithm, the probability of error is $(1 - \text{accuracy})^8$, or 69.5%.

| | True Positive | True Negative |
|---|---|---|
| Predict Positive | 1605 | 654 |
| Predict Negative | 2588 | 18660 |

**Table B.1:** Performance of the merge classifier. The MCC of this classifier is 0.453.

significantly improve the performance of tracklet merging, and could make possible a much finer-grained transaction classifier focused on the behavior of an individual rather than the aggregated behavior of customers in a store.

# Bibliography

**Anderson, C., Domingos, P. and Weld, D.** (2002). Relational Markov Models and their Application to Adaptive Web Navigation. In *Proceedings of the eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM.

**Bafna, S.** (2003). Space Syntax: A Brief Introduction to Its Logic and Analytical Techniques. *Environment and Behavior* **35**, pages 17–29.

**Bauer, D. and Kitazawa, K.** (2010). Using Laser Scanner Data to Calibrate Certain Aspects of Microscopic Pedestrian Motion Models. In W. W. F. Klingsch, C. Rogsch, A. Schadschneider and M. Schreckenberg, eds., *Pedestrian and Evacuation Dynamics 2008*. Springer Berlin Heidelberg. ISBN 978-3-642-04504-2, pages 83–94.

**Berrow, J., Beechan, J., Quaglia, P., Kagarlis, M. and Gerodimos, A.** (2005). Calibration and Validation of the Legion Simulation Model Using Empirical Data. *Pedestrian and Evacuation Dynamics*, pages 167–181.

**Bonabeau, E.** (2002). Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences* **99**, pages 7280–7287.

**Borges, J. and Levene, M.** (2000). Data mining of user navigation patterns. In B. Masand and M. Spiliopoulou, eds., *Web Usage Analysis and User Profiling*, volume 1836 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, pages 92–112.

**Borzin, A., Rivlin, E. and Rudzsky, M.** (2007). Surveillance event interpretation using generalized stochastic petri nets. In *Eighth International*

*Workshop on Image Analysis for Multimedia Interactive Services, 2007. (WIAMIS '07).*

**Bourimi, M., Mau, G., Steinmann, S., Klein, D., Templin, S., Kesdogan, D. and Schramm-Klein, H.** (2011). A privacy-respecting indoor localization approach for identifying shopper paths by using end-users mobile devices. In *Eighth International Conference on Information Technology: New Generations (ITNG), 2011*.

**Browarek, S.** (2010). *High resolution, low cost, privacy preserving human motion tracking system via passive thermal sensing*. Master's thesis, Massachusetts Institute of Technology.

**Burkhard, R., Bischof, S. and Herzog, A.** (2008). The potential of crowd simulations for communication purposes in architecture. In *Proceedings of the 2008 12th International Conference Information Visualisation (IV '08)*. Washington, DC, USA: IEEE Computer Society. ISBN 978-0-7695-3268-4.

**Burstedde, C., Klauck, K., Schadschneider, A. and Zittartz, J.** (2001). Simulation of pedestrian dynamics using a two-dimensional cellular automaton. *Physica A: Statistical Mechanics and its Applications* **295**, pages 507–525. ISSN 0378-4371.

**Carvalho, R. and Batty, M.** (2003). A rigorous definition of axial lines: ridges on isovist fields. Technical report, Centre for Advanced Spatial Analysis (University College London).

**Catledge, L. and Pitkow, J.** (1995). Characterizing browsing strategies in the world-wide web. *Computer Networks and ISDN systems* **27**, pages 1065–1073.

**Daamen, W.** (2004). *Modelling passenger flows in public transport facilities*. Ph.D. thesis, Delft University of Technology.

**Davis, J. and Goadrich, M.** (2006). The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd international conference on Machine learning*. ACM.

**DeCamp, P.** (2007). *HeadLock: Wide-Range Head Pose Estimation for Low Resolution Video*. Master's thesis, Massachusetts Institute of Technology.

**DeCamp, P., Shaw, G., Kubat, R. and Roy, D.** (2010). An Immersive System for Browsing and Visualizing Surveillance Video. In *Proceedings of the ACM International Conference on Multimedia*.

**Eastman, C.** (1971). GSP: A system for computer assisted space planning. In *Proceedings of the 8th workshop on Design automation*. ACM New York, NY, USA.

**Farley, J. U. and Ring, L. W.** (1966). A stochastic model of supermarket traffic flow. *Operations Research* **14**, pages 555–567. ISSN 0030364X.

**Fasel, B. and Luettin, J.** (2003). Automatic facial expression analysis: a survey. *Pattern Recognition* **36**, pages 259–275.

**Fleischman, M., Decamp, P. and Roy, D.** (2006). Mining temporal patterns of movement for video content classification. In *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*. ACM. ISBN 1595934952.

**Fu, Y., Guo, G. and Huang, T. S.** (2010). Age synthesis and estimation via faces: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**, pages 1955–1976.

**Gabriel, P. F., Verly, J. G., Piater, J. H. and Genon, A.** (2003). The State of the Art in Multiple Object Tracking Under Occlusion in Video Sequences. In *In Advanced Concepts for Intelligent Vision Systems (ACIVS), 2003*.

**Gipps, P. G. and Marksjö, B.** (1985). A micro-simulation model for pedestrian flows. *Mathematics and Computers in Simulation* **27**, pages 95–105. ISSN 0378-4754.

**Goffman, E.** (1966). *Behavior in public places: Notes on the social organization of gatherings*. Free Press.

**Google** (2011). Prediction API. http://code.google.com/apis/predict.

**Guadagni, P. and Little, J.** (1983). A logit model of brand choice calibrated on scanner data. *Marketing science*, pages 203–238.

**Gullichsen, E. and Chang, E.** (1985). Generative design in architecture using an expert system. *The Visual Computer* **1**, pages 161–168.

**Gupta, S.** (1988). Impact of Sales Promotions on When, What, and How Much to Buy. *Journal of Marketing Research*, pages 342–355.

**Hamuro, Y., Kawata, H., Katoh, N. and Yada, K.** (2002). A machine learning algorithm for analyzing string patterns helps to discover simple and interpretable business rules from purchase history. *Progress in Discovery Science*, pages 188–196.

**Helbing, D.** (1991). A mathematical model for the behavior of pedestrians. *Behavioral Science* **36**, 298. ISSN 00057940.

**Helbing, D., Farkas, I. and Vicsek, T.** (2000). Simulating dynamical features of escape panic. *Nature* **407**, pages 487–490.

**Helbing, D., Keltsch, J. and Molnár, P.** (1997). Modelling the evolution of human trail systems. *Nature* **388**, pages 47–50.

**Helbing, D. and Molnár, P.** (1995). Social force model for pedestrian dynamics. *Phys. Rev. E* **51**, pages 4282–4286.

**Helbing, D., Molnár, P., Farkas, I. and Bolay, K.** (2001). Self-organizing pedestrian movement. *Environment and Planning B: Planning and Design* **28**, pages 361–384.

**Henderson, L. F.** (1971). The statistics of crowd fluids. *Nature* **229**, pages 381–383.

**Hillier, B. and Hanson, J.** (1984). *The Social Logic of Space*. Cambridge University Press.

**Hillier, B., Hanson, J. and Graham, H.** (1987). Ideas are in things: an application of the space syntax method to discovering house genotypes. *Environment and Planning B: Planning and Design* **14**, pages 363–385.

**Hillier, B., Leaman, A., Stansall, P. and Bedford, M.** (1976). Space syntax. *Environment and Planning B: Planning and Design* **3**, pages 147–185.

**Hillier, B., Major, M., Desyllas, J., Karimi, K., Campos, B. and Stonor, T.** (1996). Tate Gallery, Millbank: a study of the existing layout and new masterplan proposal. Technical report, Bartlett School of Graduate Studies, University College London.

**Hong, P., Huang, T. and Turk, M.** (2000). Gesture modeling and recognition using finite state machines. In *Fourth IEEE International Conference on Automatic Face and Gesture Recognition*. IEEE Computer Society, page 410.

**Hoogendoorn, S.** (2003). Microscopic simulation of pedestrian flows. In *Transportation Research Board Annual Meeting, Washington, DC*.

**Hoogendoorn, S. and Daamen, W.** (2005). Pedestrian behavior at bottlenecks. *Transportation Science* **39**, page 147.

**Hoogendoorn, S., Daamen, W. and Bovy, P.** (2003). Extracting Microscopic Pedestrian Characteristics from Video Data. In *Transportation Research Board Annual Meeting, Washington, DC*.

**Hui, S., Bradlow, E. and Fader, P.** (2009a). Testing behavioral hypotheses using an integrated model of grocery store shopping path and purchase behavior. *The Journal of Consumer Research* **36**, pages 478–493.

**Hui, S., Fader, P. and Bradlow, E.** (2009b). The traveling salesman goes shopping: the systematic deviations of grocery paths from TSP-optimality. *Marketing Science* **28**, pages 566–572.

**Ivanov, Y., Wren, C., Sorokin, A. and Kaur, I.** (2007). Visualizing the history of living spaces. *IEEE Transactions on Visualization and Computer Graphics*, pages 1153–1160.

**Jo, J. and Gero, J.** (1998). Space layout planning using an evolutionary approach. *Artificial Intelligence in Engineering* **12**, pages 149–162.

**Kaneda, T. and Suzuki, T.** (2005). A simulation analysis for pedestrian flow management. In *Agent-Based Simulation: From Modeling Methodologies to Real-World Applications*, volume 1 of *Agent-Based Social Systems*. Springer Tokyo. ISBN 978-4-431-26925-0, pages 220–232.

**Kerridge, J., Hine, J. and Wigan, M.** (2001). Agent-based modelling of pedestrian movements: the questions that need to be asked and answered. *Environment and Planning B: Planning and Design* **28**, pages 327–342.

**Kholod, M., Nakahara, T., Azuma, H. and Yada, K.** (2010). The influence of shopping path length on purchase behavior in grocery store. In R. Setchi, I. Jordanov, R. Howlett and L. Jain, eds., *Knowledge-Based and Intelligent Information and Engineering Systems*, volume 6278 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, pages 273–280.

**Kirchner, A. and Schadschneider, A.** (2002). Simulation of evacuation processes using a bionics-inspired cellular automaton model for pedestrian dynamics. *Physica A: Statistical Mechanics and its Applications* **312**, pages 260–276. ISSN 0378-4371.

**Kitani, K., Sato, Y. and Sugimoto, A.** (2007). Recovering the basic structure of human activities from a video-based symbol string. In *IEEE Workshop on Motion and Video Computing, 2007. (WMVC '07)*.

**Kwon, Y. and Lobo, N.** (1999). Age classification from facial images. *Computer Vision and Image Understanding* **74**, pages 1–21.

**Lafferty, J., McCallum, A. and Pereira, F.** (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning (ICML)*.

**Larson, J., Bradlow, E. and Fader, P.** (2005). An exploratory look at supermarket shopping paths. *International Journal of Research in Marketing* **22**, pages 395–414.

**Larson, R.** (1987). Perspectives on Queues: Social Justice and the Psychology of Queueing. *Operations Research*, pages 895–905.

**Lavee, G., Rivlin, E. and Rudzsky, M.** (2009). Understanding video events: a survey of methods for automatic interpretation of semantic occurrences in video. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* **39**, pages 489–504. ISSN 1094-6977.

**Mäkinen, E. and Raisamo, R.** (2008). Evaluation of gender classification methods with automatically detected and aligned faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **30**, pages 541–547. ISSN 0162-8828.

**March, L. and Steadman, P.** (1971). *The Geometry of Environment*. MIT Press.

**McHugh, J., Konrad, J., Saligrama, V. and Jodoin, P.** (2009). Foreground-adaptive background subtraction. *Signal Processing Letters, IEEE* **16**, pages 390–393.

**Mehran, R., Oyama, A. and Shah, M.** (2009). Abnormal crowd behavior detection using social force model. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009*. ISSN 1063-6919.

**Meilinger, T., Franz, G. and Bülthoff, H.** (2009). From Isovists via Mental Representations to Behaviour: First Steps Toward Closing the Causal Chain. *Environment and Planning B: Planning and Design*.

**Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M. and Euler, T.** (2006). Yale: Rapid prototyping for complex data mining tasks. In L. Ungar, M. Craven, D. Gunopulos and T. Eliassi-Rad, eds., *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM. ISBN 1-59593-339-5.

**Miller, M.** (2011). *Semantic Spaces: Behavior, Language and Word Learning in the Human Speechome Corpus*. Master's thesis, Massachusetts Institute of Technology.

**Moe, W.** (2006). An empirical two-stage choice model with varying decision rules applied to internet clickstream data. *Journal of marketing research* **43**, pages 680–692.

**Moeslund, T. and Granum, E.** (2001). A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding* **81**, pages 231–268.

**Montgomery, A., Li, S., Srinivasan, K. and Liechty, J.** (2004). Modeling online browsing and path analysis using clickstream data. *Marketing Science*, pages 579–595.

**Moré, J., Garbow, B. and Hillstrom, K.** (1980). *User guide for MINPACK-1*. Argonne National Laboratory Argonne, IL.

**Moussaïd, M., Helbing, D., Garnier, S., Johansson, A., Combe, M. and Theraulaz, G.** (2009). Experimental study of the behavioural mechanisms underlying self-organization in human crowds. *Proceedings of the Royal Society B: Biological Sciences* **276**, pages 2755–2762.

**Moussaïd, M., Helbing, D. and Theraulaz, G.** (2011). How simple rules determine pedestrian behavior and crowd disasters. *Proceedings of the National Academy of Sciences* **108**, page 6884.

**Narahara, T.** (2007). *The Space Re-Actor: Walking a Synthetic Man through Architectural Space*. Master's thesis, Massachusetts Institute of Technology.

**Perkowitz, M. and Etzioni, O.** (1997). Adaptive web sites: an AI challenge. In *International Joint Conference on Artificial Intelligence*, Volume 15.

**Pittore, M., Basso, C. and Verri, A.** (1999). Representing and Recognizing Visual Dynamic Events with Support Vector Machines. *Proceedings of the* $10^{th}$ *International Conference on Image Analysis and Processing*, pages 18–23.

**Rabiner, L.** (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE* **77**, pages 257–286.

**Redner, R. A. and Walker, H. F.** (1984). Mixture Densities, Maximum Likelihood and the EM Algorithm. *SIAM Review* **26**, pages 195–239. ISSN 00361445.

**Retail Sails** (2011). Retail Sails Chain Store Productivity Report. http://retailsails.com/2011/08/23/retailsails-exclusive-ranking-u-s-chains-by-retail-sales-per-square-foot/.

**Roy, D., Patel, R., DeCamp, P., Kubat, R., Fleischman, M., Roy, B., Mavridis, N., Tellex, S., Salata, A., Guinness, J., Levit, M. and Gorniak, P.** (2006). The Human Speechome Project. In *Proceedings of the 28th Annual Cognitive Science Conference.*.

**Senior, A., Hampapur, A., Tian, Y.-L., Brown, L., Pankanti, S. and Bolle, R.** (2006). Appearance Models for Occlusion Handling. *Image and Vision Computing* **24**, pages 1233–1243. ISSN 0262-8856.

**Shakhnarovich, G., Viola, P. A. and Moghaddam, B.** (2002). A unified learning framework for real time face detection and classification. *Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*.

**Shaw, G.** (2010). Efficient Multiple Object Tracking Using Motion Features. Technical report, MIT Media Lab.

**Shaw, G.** (2011). *A Taxonomy of Situated Language in Natural Contexts*. Master's thesis, Massachusetts Institute of Technology.

**Siskind, J.** (2000). Visual event classification via force dynamics. In *Proceedings of the National Conference on Artificial Intelligence*. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999.

**Sminchisescu, C., Kanaujia, A. and Metaxas, D.** (2006). Conditional models for contextual human motion recognition. *Computer Vision and Image Understanding* **104**, pages 210–220. ISSN 1077-3142.

**Smith, L., Beckman, R., Baggerly, K., Anson, D. and Williams, M.** (1995). TRANSIMS: Transportation analysis and simulation system. Technical report, Los Alamos National Lab., NM.

**Sorensen, H.** (2003). Shopping environment analysis system and method with normalization. US Patent 7,606,728 B2; App. 20,040/111,454.

**Sorensen, H.** (2009). *Inside the mind of the shopper: The science of retailing*. Pearson Prentice Hall.

**Srivastava, J., Cooley, R., Deshpande, M. and Tan, P.-N.** (2000). Web usage mining: discovery and applications of usage patterns from web data. *ACM SIGKDD Explorations Newsletter* **1**, pages 12–23. ISSN 1931-0145.

**Stiny, G. and Mitchell, W.** (1978). The Palladian grammar. *Environment and Planning B: Planning and Design* **5**, pages 5–18.

**Terano, T., Kishimoto, A., Takahashi, T., Yamada, T. and Takahashi, M.** (2009). Agent-based in-store simulator for analyzing customer behaviors in a super-market. In J. Velásquez, S. Ríos, R. Howlett and L. Jain, eds., *Knowledge-Based and Intelligent Information and Engineering Systems*, volume 5712 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, pages 244–251.

**Turaga, P., Chellappa, R., Subrahmanian, V. and Udrea, O.** (2008). Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video Technology* **18**, pages 1473–1488.

**Turner, A., Doxa, M., O Sullivan, D. and Penn, A.** (2001). From isovists to visibility graphs: a methodology for the analysis of architectural space. *Environment and Planning B: Planning and Design* **28**, pages 103–122.

**Turner, A. and Penn, A.** (2002). Encoding natural movement as an agent-based system: an investigation into human pedestrian behaviour in the built environment. *Environment and Planning B: Planning and Design* **29**, pages 473–490.

**Turner, A., Penn, A. and Hillier, B.** (2005). An algorithmic definition of the axial map. *Environment and Planning B: Planning and Design* **32**, pages 425–444.

**Underhill, P.** (2000). *Why we buy: The science of shopping*. Simon & Schuster.

**Uotila, V. and Skogster, P.** (2007). Space management in a DIY store analysing consumer shopping paths with data-tracking devices. *Facilities* **25**, pages 363–374.

**US Census Bureau** (2011a). Estimated Annual Gross Margin of U.S. Retail Firms by Kind of Business: 1998 Through 2009. http://www2.census.gov/retail/releases/current/arts/gm.pdf.

**US Census Bureau** (2011b). Estimated Annual Sales of U.S. Retail and Food Services Firms by Kind of Business: 1998 Through 2009. http://www2.census.gov/retail/releases/current/arts/sales.pdf.

**US Census Bureau** (2011c). Estimated U.S. Per Capita Retail Sales by Selected Kind of Business: 2000 Through 2009. http://www2.census.gov/retail/releases/current/arts/percap.pdf.

**Vassilakis, H., Howell, A. and Buxton, H.** (2002). Comparison of Feedforward (TDRBF) and Generative (TDRGBN) Network for Gesture Based Control. *Gesture and Sign Language in Human-Computer Interaction*, pages 87–104.

**Wiener, J. and Franz, G.** (2004). Isovists as a means to predict spatial experience and behavior. In *Spatial Cognition IV-Reasoning, Action, Interaction. International Conference Spatial Cognition*, volume 3343. Springer.

**Willis, A., Gjersoe, N., Havard, C., Kerridge, J. and Kukla, R.** (2004). Human movement behaviour in urban spaces: implications for the design and modelling of effective pedestrian environments. *Environment and Planning B: Planning and Design* **31**, pages 805–828.

**Yada, K.** (2011). String analysis technique for shopping path in a supermarket. *Journal of Intelligent Information Systems* **36**, pages 385–402. ISSN 0925-9902. 10.1007/s10844-009-0113-8.

**Yada, K., Motoda, H., Washio, T. and Miyawaki, A.** (2006). Consumer behavior analysis by graph mining technique. *New Mathematics and Natural Computation (NMNC)* **2**, pages 59–68.

**Yang, T., Li, S. Z., Pan, Q. and Li, J.** (2005). Real-Time Multiple Objects Tracking with Occlusion Handling in Dynamic Scenes. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **1**, pages 970–975. ISSN 1063-6919.

**Yilmaz, A., Javed, O. and Shah, M.** (2006). Object tracking: A survey. *ACM Computing Surveys (CSUR)* **38**, 13.

**Zelnik-Manor, L. and Irani, M.** (2006). Statistical Analysis of Dynamic Actions. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **28**, pages 1530–1535.