Natural Language and Spatial Reasoning

by

Stefanie Anne Tellex

S.B., Computer Science, Massachusetts Institute of Technology (2002)
M.Eng., Computer Science, Massachusetts Institute of Technology (2003)
M.S., Media Arts and Science, Massachusetts Institute of Technology (2006)

Submitted to the Program in Media Arts and Science
School of Architecture and Planning
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2010

Author ....
Program in Media Arts and Science
School of Architecture and Planning
September, 2010

Certified by ....
Deb Roy
Associate Professor
Program in Media Arts and Sciences
Thesis Supervisor

Accepted by ....
Pattie Maes
Associate Academic Head
Program in Media Arts and Sciences

Natural Language and Spatial Reasoning

by

Stefanie Anne Tellex

Submitted to the Program in Media Arts and Science
School of Architecture and Planning
in September, 2010, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Making systems that understand language has long been a dream of artificial intelligence. This thesis develops a model for understanding language about space and movement in realistic situations. The system understands language from two real-world domains: finding video clips that match a spatial language description such as "People walking through the kitchen and then going to the dining room" and following natural language commands such as "Go down the hall towards the fireplace in the living room." Understanding spatial language expressions is a challenging problem because linguistic expressions, themselves complex and ambiguous, must be connected to real-world objects and events. The system bridges the gap between language and the world by modeling the meaning of spatial language expressions hierarchically, first capturing the semantics of spatial prepositions, and then composing these meanings into higher level structures. Corpus-based evaluations of how well the system performs in different, realistic domains show that the system effectively and robustly understands spatial language expressions.

Thesis Supervisor:
Deb Roy
Associate Professor
Program in Media Arts and Sciences

Natural Language and Spatial Reasoning

by

Stefanie Anne Tellex

The following people have served as readers for this thesis:

....

Boris Katz
Principal Research Scientist
MIT Computer Science and Artificial Intelligence Laboratory

....

Yuri Ivanov
Senior Research Scientist
Mitsubishi Electric Research Laboratories

....

Cynthia Breazeal
Associate Professor
Program in Media Arts and Sciences

Acknowledgments

There's no way I could have made it this far on my own. Most of all, I want to thank my fiancé, Piotr Mitros. He convinced me to go to graduate school in the first place. He has been my rock throughout this process. I couldn't have done it without him.

Much of this work was done in collaboration with Tom Kollar, especially spatial description clauses, the probabilistic model, the route instruction system, and the planning mechanism by forward search. Working with him has made the work much more than it would have been alone, and it has truly been a privilege.

My advisor, Deb Roy, and my committee have given me great advice throughout the process. Deb has always held me to a high standard and provided essential resources to accomplish this research. Yuri Ivanov's practical and compassionate advice helped me stay optimistic. Boris Katz has been my mentor for years; among many pieces of great advice, he taught me to always look at where the system breaks, and say why. Cynthia Breazeal leads the MURI project that funded most of this research; she has been supportive throughout.

Gerry Sussman always has his office door open. From stories about working with Terry Winograd on Shrdlu to bike rides and sailing trips, he has been both an advisor and a friend. Nick Roy has been generous with his time and his robots, going out of his way to make me feel welcome and included in his group.

Three friends in particular have helped me ride the emotional roller coaster that is graduate school. Dave has been my friend since we built .coms together in the late 90's. He helped me to appreciate what I have, and to remember that life is not a thesis. Kai-yuh showed me that I was afraid and taught me to have courage and compassion. Gremio taught me to never stop trying, even if sometimes you have to change what you're trying for.

My family has been my rock throughout this process. My grandmother, Grace Torregrossa taught me how to spell "Mississippi" when I was young and spawned a life-long interest in language. Today she doesn't always remember how wonderful and wise she is, but her family does. My mom and dad don't really understand

why I went to graduate school, but they have always been supportive. It has been wonderful living near my sister Shannon and her husband Jon. From emergency trips to Rochester when a family member is sick to convincing Mom to get into a canoe, she's been a wonderful sister and a friend. We were so proud of my brother Scott when he bought a house in Rochester; it has been wonderful to see his growing independence. And my whole family is so excited to learn that my sister Staci is pregnant: my parents' very first grandchild is due in January.

Piotr's family has been amazing. Ania and Seth are great mentors and models for how to have both a career and a baby. Their son Moby is awesome. Mama and Tata have put up with me working when they visit and have been generous with advice.

My best friends Carie and Amy and I graduated from Irondequoit High School together. Amy has a Ph.D. in Chemistry and Carie is finishing up her Ph.D. in Astronomy. Now they are in my wedding. Spotty lives!

Our neighbors Doug and Jenny have been one of the great things about living in Cambridge. Having friends nearby is awesome, from the delicious dinners, the innumerable games of Mario Kart, to the time when Piotr's mom got locked out with the stove on and they rescued her while I was off at work on a paper deadline to liquid nitrogen ice cream. Jenny's mom, Susan Akana, has been a great mentor as I embarked on the career choices, providing me impartial and compassionate guidance.

My groupmates in the Cognitive Machines group are always willing to read paper drafts, offer advice, have interesting discussions, and listen to venting. The Media Lab itself is like nowhere else in the world, and I've learned so much from the people here. I'm especially grateful to Necsys, who has always been generous with their time and resources, from finding random pieces of equipment to answering questions, to helping us run a gigantic disk array for the Human Speechome Project. Our group administrators have been fantastic; Karina and Mutsumi put in a tremendous amount of effort to make everything easy and painless for us. Finally, I'd like to thank Linda Peterson, who has been a compassionate and nurturing advocate.

And finally, I want to finish where I started, by thanking my fiancé, Piotr Mitros. I wouldn't have gotten anywhere without him.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

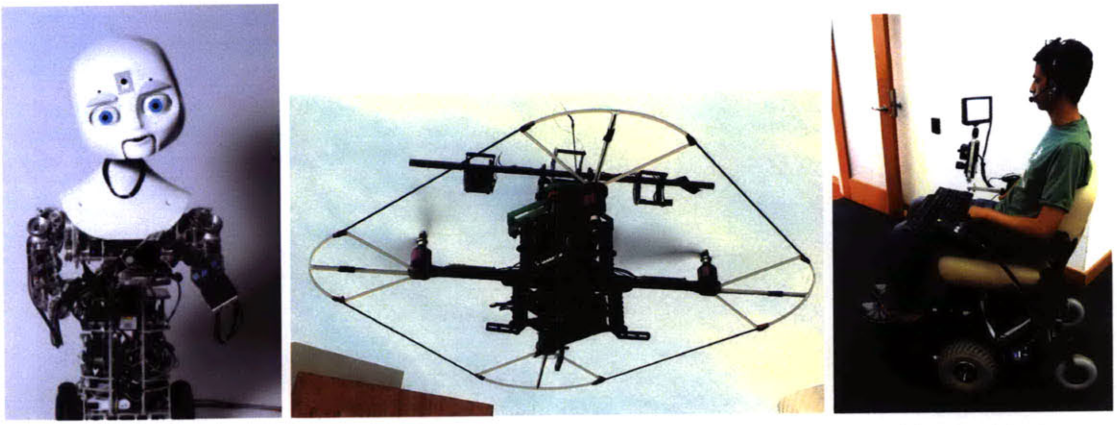
> When we study human language, we are approaching what some might call "the human essence," the distinctive qualities of mind that are, so far as we know, unique to man and that are inseparable from any critical phase of the human experience, personal or social.
>
> – *Noam Chomsky*

Building a computer that can understand natural language has been a dream of artificial intelligence since the Turing test was first proposed in 1950. Language is uniquely human: through language humans unify and integrate many disparate abilities. One fruitful strategy when faced with a large problem is to divide it into smaller subproblems. This approach has been hugely successful, leading to amazing progress in individual problem areas of artificial intelligence, from web search to autonomous robotic navigation. However there has been less work towards integrating results from different subfields into a consistent and coherent framework. Models from computational linguistics often model only words, and not the non-linguistic components of semantics. In this thesis I will divide the problem of language understanding not horizontally, but vertically. In other words, I will focus on a narrow subset of language, *grounding* that language in data collected from a real world. This strategy has two benefits. First, it decreases the scope of the language understanding problem, making it more tractable. Second, by choosing a semantically deep core domain, it offers an opportunity to explore the connection between linguistic and non-linguistic concepts. A model that spans domains may be more powerful and generalizable to new situations and contexts, a key challenge in creating intelligent systems.

When pursuing this strategy, the choice of sub domain is key. In this thesis I developed systems that understand spatial language. Reasoning about movement and space is a fundamental competence of humans and many animals. Humans use spatial language to tell stories and give directions, abstracting away the details of a complex event into a few words such as "across the kitchen." A system that understands spatial language could be directly useful to people by finding video that matches spatial language descriptions, or giving natural language directions. In order to reason in this domain, the system must somehow map information from language into the world. This thesis describes a model for one way that this mapping could be carried out and applies the model to two realistic domains.

Applying the model to a realistic domain is important to show that it works. Working with more than one domain means the model must be more generalizable, making it more likely the ideas will apply in a wide array of contexts. Moreover, a system that works in a realistic domain is useful in its own right, making it more likely that the ideas behind it will have impact. The choice of domains depends on several factors. First, the applications should be compelling. Second, the scope of the problem should be neither too small nor too large. Finally, the two domains should be similar enough to reuse the same modeling machinery, but different enough to require interesting generalizations. In this thesis I focus on natural language video retrieval and direction giving and understanding for robots.

Video retrieval is a compelling application: in the United States alone, there are an estimated 30 million surveillance cameras installed, which record four billion hours of video per week [Vlahos, 2008]. Analyzing and understanding the content of video data remains a challenging problem. A spatial language interface to video data can help people naturally and flexibly find what they are looking for in video collections. For example, a system installed in a store could help analysts design better layouts



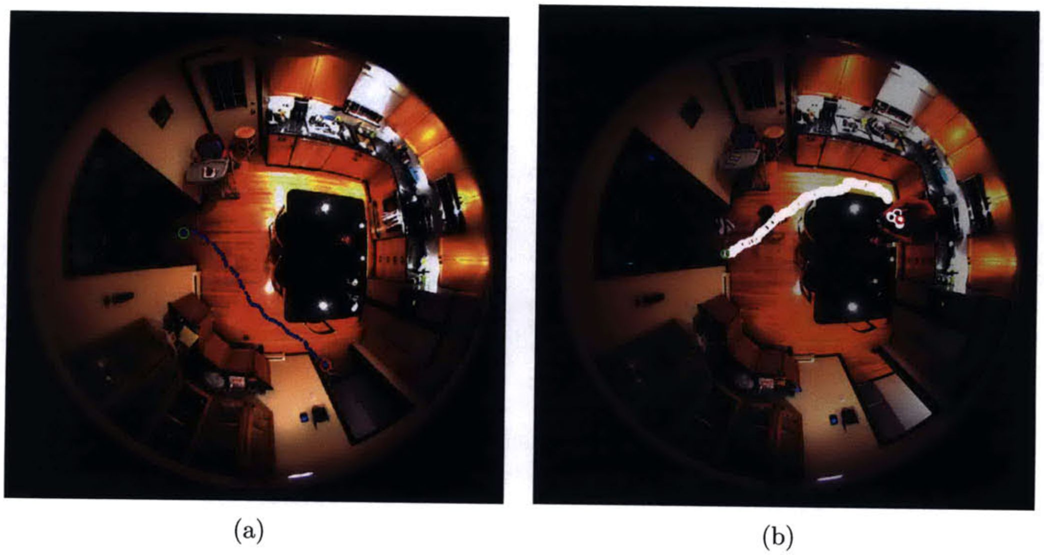
(a) humanoid (b) helicopter (c) wheelchair

Figure 1-1: Robot platforms used in this thesis.

and understand shopper behavior.

Studying language used to give directions could enable a robot to understand natural language directions. People talk to robots even if they do not have microphones installed [Kinzer, 2009], and it makes sense to build systems that understand what they say. A robot that understands natural language is easy for anyone to use without special training. In complex environments, human operators may not have the cognitive or sensory resources to use keyboard-and-mouse interfaces. Natural language keeps the hands and eyes free for other tasks. Figure 1-1 shows platforms that have used the natural language interface described in part in this thesis.

Both these domains require a system that can connect language to concrete non-linguistic contexts. To explore these connections, I obtained corpora of language where each utterance was paired with non-linguistic contextual information. In the case of video retrieval, I have collected a corpus of natural language descriptions of video clips. The video corpus consists of data recorded from a fish-eye camera installed in the ceiling of a home. Sample frames from the video corpus, retrieved by the system for the query "across the kitchen," are shown in Figure 1-2. To associate video clips with a natural language description, annotators were asked to write a short phrase describing the motion of a person in the clip. Using this corpus of descriptions paired with video clips, I trained models of the meanings of some spatial prepositions and explored their semantics by analyzing which features are most important for



(a)

(b)

Figure 1-2: Frames from two clips returned for the query "across the kitchen."

good classification performance. I implemented systems for two versions of this task: a restricted corpus limited to one prepositional phrase for each video clip, and an open-ended corpus in which annotators described a person's movement in video using whatever language seemed most natural to them. Figure 1-3 shows the most frequent descriptions that appeared in the restricted corpus. These corpora provide a way to train and evaluate models of spatial semantics in a restricted, but still realistic context.

For robotic language understanding, I worked with a corpus collected by Nicholas Roy and his students at MIT CSAIL [Kollar et al., 2010]. Subjects were given a tour of a room and asked to write down directions between two locations inside a building. The corpus consists of paragraph-length route instructions. One set of directions from the corpus is shown in Figure 1-4. In order to enable an offline evaluation, the dataset includes a log of a robot's observations of the environment. The dataset was collected by a mobile robot instrumented with a LIDAR and camera. The log thus contains odometry, laser scans, and camera images. A SLAM module used the laser scans and odometry to create a map as the robot explored the environment [Grisetti et al., 2007], while the camera detected baseline objects in order to create a semantic

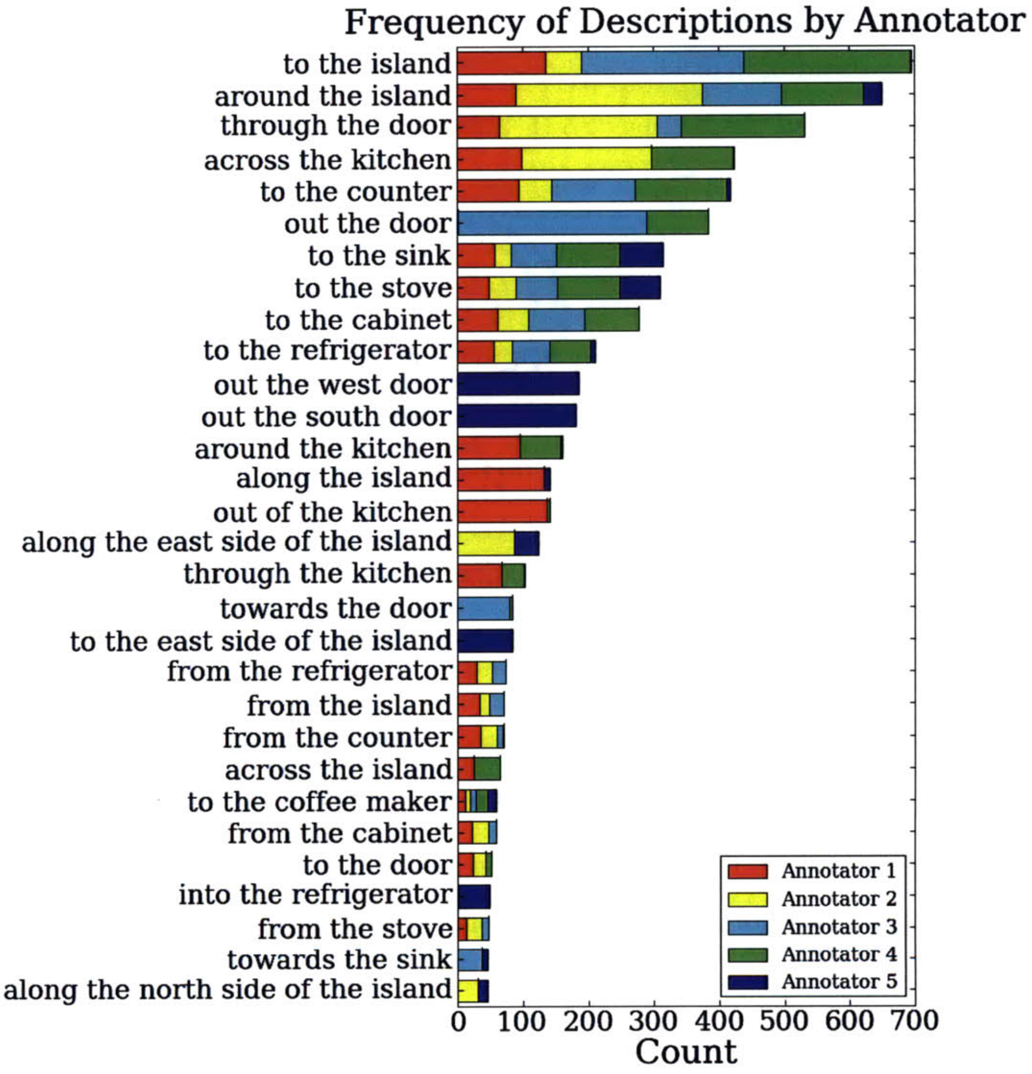


Figure 1-3: Histogram of descriptions of video clips. Annotators watched a several second long clip and completed the sentence "The person is going ..." with one prepositional phrase.

> With your back to the windows, walk straight through the door near the elevators. Continue to walk straight, going through one door until you come to an intersection just past a white board. Turn left, turn right, and enter the second door on your right (sign says "Administrative Assistant").

Figure 1-4: A typical set of directions from the corpus described in Chapter 5.

map. The log enables testing models at following directions from the corpus offline, without the overhead of deploying a physical robot in a user study. This data offers an opportunity to map the language to the nonlinguistic spatial context.

Spatial language is modeled as a sequence of structured clauses called *spatial description clauses* or SDCs, developed jointly with Thomas Kollar and introduced in this thesis. SDCs consist of a *figure*, a *verb*, a *spatial relation* and a *landmark*. The SDCs for the sentence "Walk down the hall towards the elevators" are [V: Walk, SR: down, L: the hall] and [SR: towards, L: the elevator.] Here, the figure is an implied "you." SDCs are automatically extracted from input text using a conditional random field chunker. Language is modeled as an independent sequence of SDCs using a novel probabilistic model, with factors corresponding to the fields of the SDC. This model allows each component of the SDC to be grounded separately and combined back together to ground spatial language discourse.

Overall the system performs quite well. For direction understanding, the system correctly follows between 50% and 60% of directions in the corpus, compared to 85% for human performance. Furthermore, spatial relations improve performance when using only local information in the map to follow the directions. When performing video retrieval with the open-ended corpus, the system effectively retrieves clips that match natural language descriptions: 58.3% were ranked in the top two of ten in a retrieval task, compared to 39.9% without spatial relations.

The rest of this document is organized as follows: Chapter 2 reviews related work. Chapter 3 describes how to model the meanings of spatial prepositions in isolation. Chapter 4 describes how to compose these individual meanings to model spatial language discourse. Chapter 5 describes how this model is used to follow natural language route instructions through realistic environments. Chapter 6 applies the model to spatial language video retrieval. Chapter 7 describes extensions to the model that allow it to model spatio-temporal verbs such as “follow” and “meet.” Finally, Chapter 8 concludes with ongoing and future work and contributions.

Chapter 2

Related Work

This thesis develops a framework for understanding spatial language. Related work includes work from cognitive science about the spatial reasoning and work from cognitive semantics about the linguistics of spatial language. In addition, many in artificial intelligence have worked directly on route direction understanding and video search.

2.1 Cognitive Semantics

There is a long history of systems that understand natural language in small domains, going back to Winograd [1970] and his work building a natural language interface to a robot that manipulated blocks in a virtual environment. This thesis builds on previous work by bringing the system in contact with realistic data: it is trained and evaluated using data from a corpus of naive annotators who are not otherwise involved with the development of the system. The language is about real situations, describing routes in office buildings and people's movement. The system must tackle unsanitized language not tuned and filtered by author of the system. A constant theme in the work is the struggle to balance open-ended natural language understanding with the limitations arising from the sensing and understanding capabilities of the system.

The linguistic structure extracted from spatial language expressions and many of the features in the model for spatial relations are based on the theories of Jackendoff [1983], Landau and Jackendoff [1993] and Talmy [2005]. This work aspires to be a computational instantiation of some of the ideas in their theories. For example, Talmy [2005] says that for a figure to be across a particular ground, among other things, the axes of the two objects must be "roughly perpendicular." The implementation of "across" in this work extends his definition by giving an algorithm for computing the axes a figure imposes on a ground, and a set of features which quantify "roughly perpendicular," using a machine learning algorithm to fine-tune the distinctions by training on labeled data.

Others have implemented and tested models of spatial semantics. Regier [1992] built a system that assigns labels such as "through" to a movie showing a figure moving relative to a ground object. Bailey [1997] developed a model for learning the meanings of verbs of manipulation such as "push" and "shove." Kelleher and Costello [2009] and Regier and Carlson [2001] built models for the meanings of static spatial prepositions such as "in front of" and "above." Building on their paradigm of testing the semantics of spatial prepositions against human judgements, this work focuses on realistic situations, requiring the model to be robust to noise, and enabling an analysis of how the semantics of spatial prepositions change in different real-world domains.

Siskind [2001] created a system for defining meanings for words such as "pick up" and "put down." His framework defined words in terms of *event logic*, a formalism that is an extension of Allen relations [Allen, 1983]. The framework reasons about formal temporal relations between primitive force-dynamic properties such as "supports" and "touches" and uses changes in these properties to define meanings for verbs. His framework focuses on word-level event recognition and features. In contrast, the work described in this thesis describes more open-ended types of spatial language, and for composing individual word models together.

2.2 Applications

This section reviews work related to video retrieval, route direction understanding, and GIS applications.

2.2.1 Video Search

Katz et al. [2004] built a natural language interface to a video corpus which can answer questions about video, such as "Show me all cars leaving the garage." Objects are automatically detected and tracked, and the tracks are converted into an intermediate symbolic structure based on Jackendoff [1983] that corresponds to events detected in the video. My work focuses on handling complex spatial prepositions such as "across" while they focus on understanding a range of questions involving geometrically simpler prepositions.

Fleischman et al. [2006] built a system that recognizes events in video recorded in the kitchen. Their system learns hierarchical patterns of motion in the video, creating a lexicon of patterns. The system uses the lexicon to create feature vectors from video events, which are used to train a classifier that can recognize events in the video such as "making coffee." The system also uses classifiers to recognize events but focuses on events that match natural language descriptions rather than finding higher level patterns of activity.

Black et al. [2004] describe a system that tracks objects in video surveillance data and stores the results in a database for analysis and retrieval. Their database stores the trajectories of moving objects as they move between multiple nearby cameras. The output of the vision system is a database of three dimensional motion trajectories of moving objects. Next, a meta-data layer produces a semantic description of an object's activity, including entry and exit point, along with routes taken through the field of view and time spent at each node. They focus on building a system that can retrieve object motions in regions of the video. My work extends this idea to add a natural language interface on top of a database of object trajectories.

More generally, Naphade et al. [2006] describe the Large-Scale Concept Ontology for Multimedia (LSCOM), an effort to create a taxonomy of concepts that are automatically extractable from video, that are useful for retrieval, and that cover a wide variety of semantic phenomena. Retrieval systems such as the one described by Li et al. [2007] automatically detect these concepts in video and map queries to the concepts in order to find relevant clips. This work describes a complementary effort to recognize fine-grained spatial events in video by finding movement trajectories that match a natural language description of motion.

Researchers have developed video retrieval interfaces using non-linguistic input modalities which are complementary to linguistic interfaces. Ivanov and Wren [2006] describe a user interface to a surveillance system that visualizes information from a network of motion sensors. Users can graphically specify patterns of activation in the sensor network in order to find events such as people entering through a particular door. Yoshitaka et al. [1996] describe a query-by-example video retrieval system that allows users to draw an example object trajectory, including position, size, and velocity, and finds video clips that match that trajectory. The natural language query interface that is developed in this work complements these interfaces: language can succinctly express paths such as "towards the sink," which would need to be drawn as many radial lines to be expressed graphically, while queries for complicated, intricate trajectories can be drawn. Furthermore, queries expressed as text strings are easily repeatable; in contrast, it is difficult to draw (or tell someone else to draw) the exact same path twice in a pen-based system. The combination of a pen-based interface and a natural language interface is more powerful than either interface on its own.

2.2.2 Understanding Natural Language Directions

Many authors have proposed formalisms similar in spirit to spatial description clauses for reasoning about the semantics of natural language directions. Many of these representations are more expressive than SDCs, but correspondingly more difficult to automatically extract from text, to the point where many authors sidestep this problem by using human annotations. SDCs capture many of the semantics of natural language directions, while still being simple enough to extract and reason about automatically.

For example, Levit and Roy [2007] describes a probabilistic model for finding likely paths described by dialogs from the MapTask corpus [Thompson et al., 1993], which consists of two people engaging in dialog about a path drawn on a map. Semantic units called navigational informational units (NIUs) are annotated in the text, and the system finds paths given a sequence of NIUs. This formulation is the most similar to SDCs of the frameworks reviewed here. For a phrase like “move two inches towards the house,” an NIU contains a path descriptor (“move... towards”), a reference object (“the house”), and a quantitative description (“two inches”). Spatial description clauses break down the instruction in a similar way, separating the path descriptor into a verb and spatial relation, but not explicitly modeling the quantitative description, since it appears so infrequently in the corpus. The possible path descriptors of their formalism correspond to spatial relations in the current framework. The SDC formalism explicitly represents the structure common to any spatial referring expression, whether it refers to a position, an orientation, a move, or a compound reference to a landmark such as “the doors near the elevators.”

Bugmann et al. [2004] identified a set of 15 primitive procedures associated with clauses in a corpus of spoken natural language directions. Example primitives include “go,” “location is,” and “enter roundabout.” This work follows their methodology of corpus-based robotics, working with natural language directions given by a human for another human. An individual spatial description clause in the current framework could correspond to one of their primitives actions. Spatial description clauses explicitly represents a more general structure common to all of their primitives, enabling a factorization of the system into a spatial-relation processing module, and a landmark processing module, both of which can be used in other applications.

Macmahon [2006] built a system that follows natural language directions created by a human for another human through a simulated environment. His system represents each clause in a set of directions as one of four simple actions: move, turn, verify, and declare-goal. A parser extracts these simple actions from text, and forms compound actions, consisting of a simple action plus pre-conditions, while-conditions, and post-conditions. A compound action in his formalism is roughly equivalent to an SDC. This framework is more expressive than SDCs: a compound action can have more than one pre-, post-, and while-conditions. For example for “Follow the atrium all the way to the right,” “follow the atrium” can be seen as a while-condition, while "all the way to the right" describes a post-condition for the path segment. In the corpus used in this thesis, clauses involving more than one pre-, post-, or while-conditions are relatively rare in the corpus; when they occur, they are modeled as separate spatial description clauses.

Klippel et al. [2005] created a formalism for representing route knowledge called wayfinding choremes. At each decision point in the route, a possible direction to take is discritized into one of seven equidistant directions. These directions can be lexicalized as sharp right, right, half right, straight, half left, left, sharp left. (Back is a special case.) A sequence of wayfinding choremes can be chunked together to create higher-order direction elements. Like Klippel et al. [2005], the model described here discritizes orientation into one of four possible orientations. However, rather than treating turning as primitive, in the model described in this thesis, landmarks are the key feature used to connect between natural language directions and the external world. Each SDC describes a transition between two viewpoints, almost always with respect to a landmark: only 21% of the SDCs in the corpus appear without an associated landmark. Landmarks are a key part of natural language directions and are represented explicitly in the formalism described here.

Wei et al. [2009] addressed the direction understanding problem by modeling directions as a sequence of landmarks. The work in this thesis builds on the previous system by adding spatial relation understanding, rather than using landmarks alone. This enables the system to more completely model the natural language, while still exploiting the structure of landmarks in the map to follow the directions.

Matuszek et al. [2010] created a system that follows directions by learning meanings for phrases like "the third door" from a corpus of route instructions paired with routes. Their system uses a machine translation-based approach to translate from natural language to paths through the environment. The work described in this thesis takes a different approach to the problem, by imposing a semantic structure on the language (spatial description clauses), then exploiting that structure to factor the resulting probability distribution.

Dzifcak et al. [2009] created a language understanding system that simultaneously builds semantic structures representing both the goal of a sentence such as "Go to the break room," as well as the action needed to achieve that goal. A combinatory categorial grammar (CCG) parser extracts both structures from the input text. The CCG formalism enables the robot to understand complex commands going beyond following directions, such as "Go to the break room and report the location of the blue box." The work described in this thesis takes a different strategy: rather than trying to extract the entire linguistic structure from natural language input, the system described in this thesis extracts a simplified, domain-specific representation. The extraction is robust to ungrammatical sentences such as "go to hallway," and can follow directions from untrained users with high accuracy.

Others have created systems for enabling robots to understand natural language. For example, Skubic et al. [2004] described a system for enabling robots to describe their environment as in "There is a desk in front of me and a doorway behind it," as well as to obey commands like "go to the elevator." Hsiao et al. [2008] made a table-top robot that obeyed commands such as "Group the red one and the green one" using a planning-based approach. The current work focuses on spatial language describing trajectories, using a model that can robustly understand natural language commands and descriptions.

Ren et al. [2009] review video retrieval methods based on matching spatio-temporal information. They describe symbolic query languages for video retrieval, trajectory-matching approaches, and query-by-example systems.

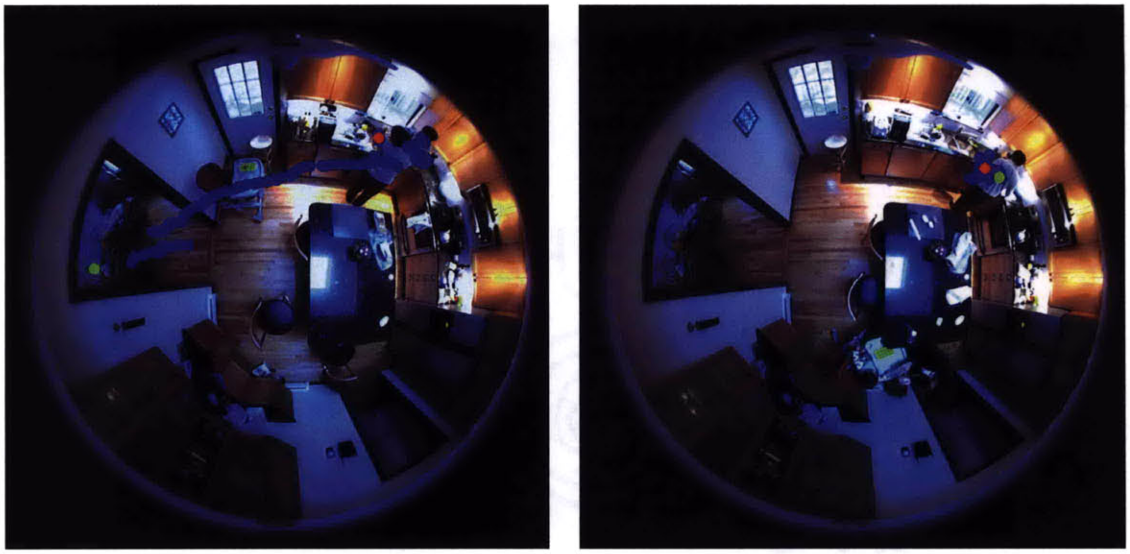
Chapter 3

The Meanings of Spatial Words

This chapter describes how the system grounds the meaning of spatial prepositions, culminating in a phrase-based spatial language understanding system. The following chapter describes how to use these models to understand more open-ended spatial language sentences and paragraphs: the thesis starts with models for words and phrases, and moves on to discourse. Spatial relations are modeled as probabilistic distributions for recognizing words paired with scenes. The distributions are trained from labeled examples using a set of geometric features that capture the semantics of spatial prepositions: the identification of these features is a key contribution of this thesis.

The distribution being modeled is the probability of a particular spatial relation sr given a trajectory t and an object in the environment o, $p(sr|t, o)$. This distribution corresponds to the probability that a spatial relation such as "across" or "to" describes a particular trajectory and landmark. These distributions are trained using labeled path examples. Continuous geometric paths are converted into a set of features motivated by theories of human spatial language [Talmy, 2005, Jackendoff, 1983, Landau and Jackendoff, 1993]. The input to the model is the geometry of the path and landmark object; the output is a probability that the spatial relation can be used to describe this scene. Probabilities output by the model for two different trajectories are shown in Figure 3-1.

To build models for spatial relations, the system needs to compute how well a



$p(\text{to}|path, sink) = 0.99$ $p(\text{to}|path, sink) = 0.10$

Figure 3-1: Probabilities output by the model for two different scenes for "to the sink."

phrase such as "past the door" describes a particular path segment, t. I conceive of spatial relations as two-argument functions that take a figure and a landmark. For dynamic spatial prepositions, the figure is a path, represented as a sequence of points, and the landmark is a point or a polygon. This schematization is illustrated in Figure 3-2.

I use binary distributions of the form $p(sr|t, o)$ to model the meaning of spatial prepositions for two reasons. First, a binary distribution can be directly used for video retrieval by simply returning all clips in order of probability. Second, binary distributions naturally capture the fact that a given situation can have more than one correct description. For example, the clip shown in Figure 3-2 could be described as "across the island" or "past the island."

3.1 Models for Spatial Prepositions

The system learns distributions for spatial relations using a naive Bayes probabilistic model. To model this distribution, I use features f_i that capture the semantics of spatial prepositions. These features are functions of the geometry of the path and

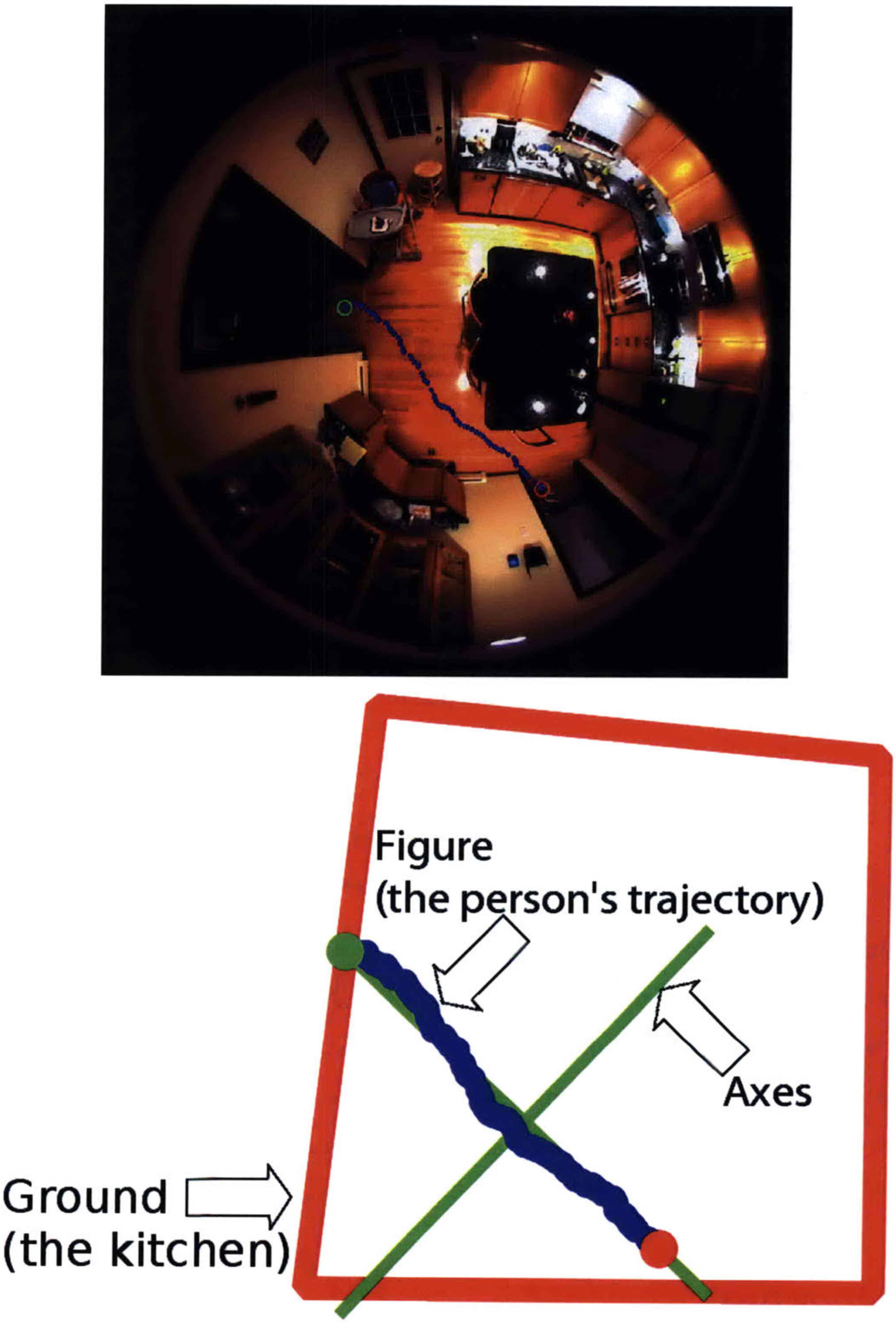


Figure 3-2: Frame from a video clip in the corpus, plus the schematic representation of the event, used for computing $p(sr|t, o)$.

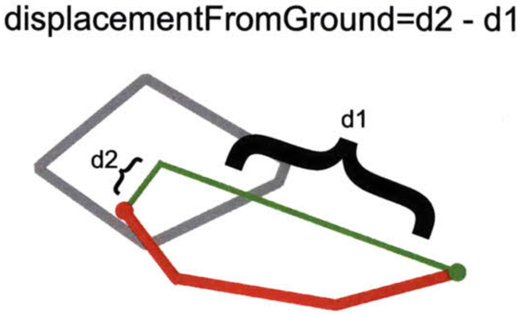


Figure 3-3: Illustration of the computation of the feature *displacementFromGround*.

landmark. For example, one of the features utilized for the spatial preposition "to" is the distance between the end of the path and the landmark's location.

$$p(sr|t,o) = p(sr|f_1 \dots f_N) \tag{3.1}$$

Next, we use Bayes' rule to rewrite the distribution, and then make the Naive Bayes assumption, that features are independent of each other.

$$p(sr|t,o) = \frac{p(f_1 \dots f_N|sr)p(sr)}{p(f_1 \dots f_N)} \tag{3.2}$$

$$p(sr|t,o) = \frac{\prod_i p(f_i|sr)p(sr)}{\sum_{sr_k} p(f_i|sr_k)p(sr_k)} \tag{3.3}$$

In this section I describe the features used for each spatial relation. The features were selected from a large set of candidate features based on how well they captured the semantics of the particular spatial relation. Some features are reused for more than one spatial relation. In this case the feature definition is repeated twice for clarity: in this way the reader can look up a word and find the complete set of features used for that word without having to flip to other parts of the document.

3.1.1 across

An important underlying concept inherent in the meaning of many spatial prepositions is the idea of coordinate axes. "Across" has been defined as a spatial relation that takes a linear figure and planar ground, and requires the figure to be perpendicular to the major axis of the ground [Talmy, 2005, Landau and Jackendoff, 1993]. However this definition does not specify how to find the major axis of the ground. In many contexts, there is no single set of axes: for example, there are many paths across a square room. The system solves this problem by finding the unique axes that the figure imposes on the ground, and then quantifying how well those axes match the ground. These axes are computed by finding the line that connects the first and last point in the figure, and extending this line until it intersects the ground. This computation is illustrated in Figure 3-2. The origin of the axes is the midpoint of this line segment, and the endpoints are the two points where the axes intersect the ground. Once the axes are known, the system computes features that capture how well the figure follows the axes, and how well the axes fit the ground.

- **figureCenterOfMassToGroundCentroid**: The distance between the center of mass of the figure and the centroid of the ground.
- **distAlongGroundBtwnAxes**: The distance along the ground between the start and end of the minor axis.
- **ratioFigureToAxes**: The ratio between the distance between the start and end points of the figure and the axes.

3.1.2 along

The axes for "along" is computed by taking a subset of the boundary of the ground object. The start of the axes is the point on the ground closest to the start of the figure; the end of the axes is the point on the ground closest to the end of the figure. The axes itself is the boundary of the ground between these two points.

- **peakDistanceToAxes**: The maximum distance between the figure and the axes it imposes, for the part of the figure that is inside the ground.
- **standardDeviation**: The standard deviation of the distance between the figure and the ground.

- **averageDistStartGroundDistEndGround**: The average of distEndGround and distStartGroundBoundary.

- **angleBetweenLinearizedObjects**: The angular difference of the slope of a line fit to the figure and the ground.

3.1.3 around

- **averageDistStartGroundDistEndGround**: The average of distEndGround and distStartGroundBoundary.

- **distStartGroundBoundary**: The distance of the start of the figure to the ground.

- **figureStartToEnd**: The distance between the start of the figure and the end.

3.1.4 away from

As in "walk away from the door."

- **displacementFromGround**: This feature measures the net distance traveled by the figure towards the ground. It is illusrated in Figure 3-3.

- **distFigureEndToGround**: The distance from the end point of the figure to the ground.

- **distFigureStartToGround**: The distance from the start point of the figure to the ground.

3.1.5 down

As in "down the hall" or "down the road."

- **standardDeviation**: The standard deviation of the distance between the figure and the ground.

- **figureCenterOfMassToAxesOrigin**: The distance between the center of mass of points in the figure and the axes origin.

- **distAlongGroundBtwnAxes**: The distance along the ground between the start and end of the minor axis.

- **eigenAxesRatio**: The ratio between the eigenvectors of the covariance matrix of the ground when represented as an occupancy grid.

3.1.6 out

- **figureCenterOfMasssToGroundCentroid**: The distance between the center of mass of the figure and the centroid of the ground.

- **averageDistance**: The average distance between the figure and the axes.

- **displacementFromGround**: This feature measures the net distance traveled by the figure towards the ground. It is illustrated in Figure 3-3.

3.1.7 past

- **distFigureStartToGround**: The distance from the start point of the figure to the ground.

- **distFigureEndToGround**: The distance from the end point of the figure to the ground.

- **distFigureStartToGround**: The distance from the start point of the figure to the ground.

- **angleFigureToAxes**: The angle between the linearized figure and the line perpendicular to the closest point on the ground.

- **axesLength**: The length of the axes, normalized.

3.1.8 through

Features for "towards" are borrowed from "across."

- **figureCenterOfMassToGroundCentroid**: The distance between the center of mass of the figure and the centroid of the ground.

- **distAlongGroundBtwnAxes**: The distance along the ground between the start and end of the minor axis.

- **ratioFigureToAxes**: The ratio between the distance between the start and end points of the figure and the axes.

3.1.9 to

- **distFigureEndToGround**: The distance from the end point of the figure to the ground.

- **displacementFromGround**: This feature measures the net distance traveled by the figure towards the ground. It is illustrated in Figure 3-3.

3.1.10 towards

- **distFigureEndToGround**: The distance from the end point of the figure to the ground.

- **displacementFromGround**: This feature measures the net distance traveled by the figure towards the ground. It is illustrated in Figure 3-3.

3.1.11 until

- **distFigureEndToGround**: The distance from the end point of the figure to the ground.

- **startPointsInGroundBoundingBox**: Is the starting point of the ground inside the figure's bounding box?

3.2 Performance in Schematic Domain

The distributions are trained and evaluated on a corpus of examples of each spatial relation. In this evaluation I report the performance on this test set overall, and for each individual feature. The system learns the distribution in Equation 3.3 from a dataset created by hand-drawing examples of paths that matched a natural language description such as "through the door." In this data set, positive examples of one spatial relation were taken to be negative examples of others. Some pairs of spatial relations, such as "to" and "towards," which are very similar, were excluded from each other's training sets. I trained classifiers for eleven spatial prepositions: "across," "along," "through," "past," "around," "to," "out," "towards," "down," "away from," and "until." "Until" is not a spatial relation in general, but I modeled it as one here because it almost always refers to an arrival event in the direction understanding corpus, as in "until you come to an intersection just past a whiteboard."

The resulting distribution is treated as a classifier for a particular spatial relation, and I report performance on a held-out test set from in Figure 3-4. This figure shows that a high-accuracy classifier was learned for each spatial relation.

The distributions can be used for information retrieval by searching a dataset of trajectories and landmarks for high and low scoring examples. Figure 3-20 shows the results of this search over a database of trajectories in a topological map of an environment completely separate from the training set. This figure shows that the system is correctly recognizing good and bad examples for each spatial relation.

Many features involve distances. However spatial relations are scale invariant: a snail can crawl across a leaf, and a spaceship can travel across the solar system. Despite the huge difference in scale, "across" can be used to describe both trajectories. This problem is solved by normalizing all distances by the size of either the ground, the figure, or the bounding box of the scene.

Finally, a key contribution of this work is to analyze the contribution of individual features to the performance of the full model. In some cases this analysis lead to just one or two features that work well for a particular spatial relation; in other cases

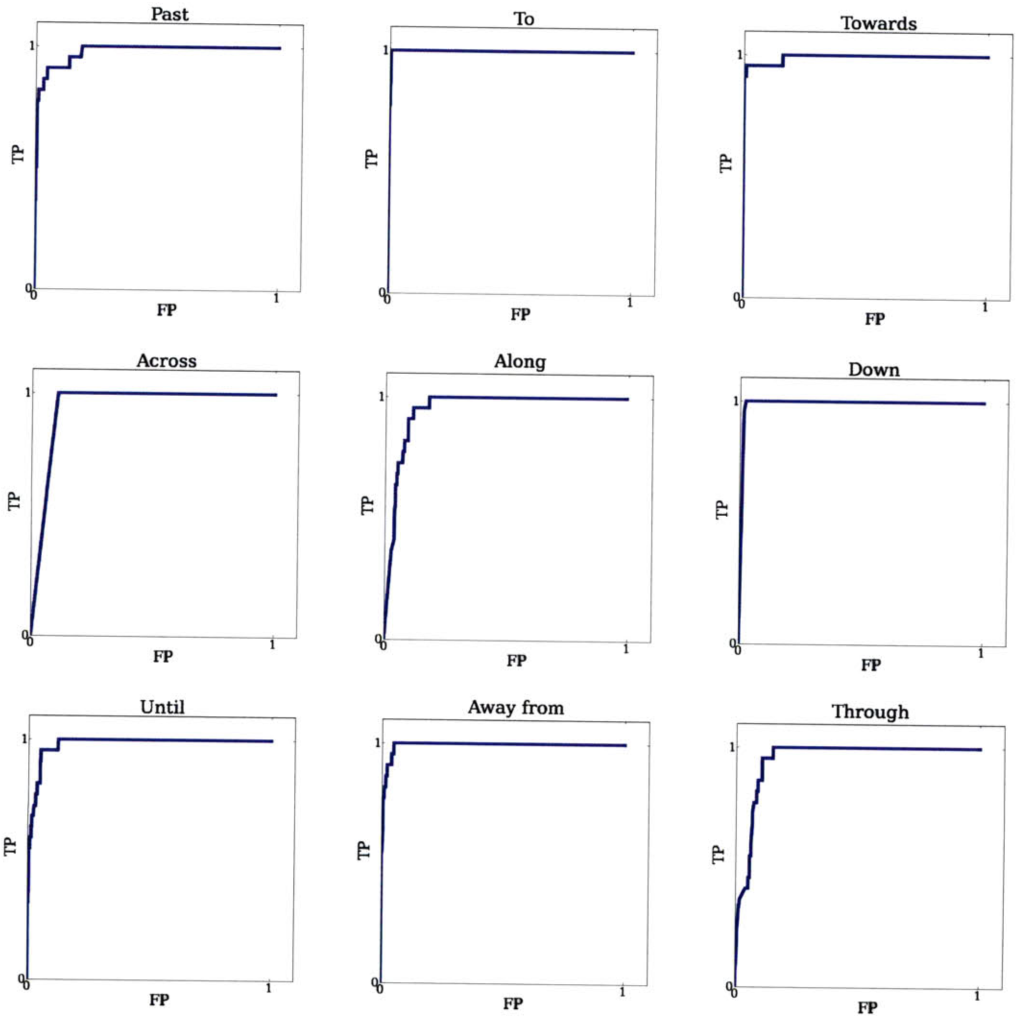


Figure 3-4: The performance on the corpus of examples drawn for a particular path, when treating the spatial relation probability as a classifier. On the horizontal axis of the ROC curve is the false positive rate (FP) and on the vertical axis is the true positive rate (TP).

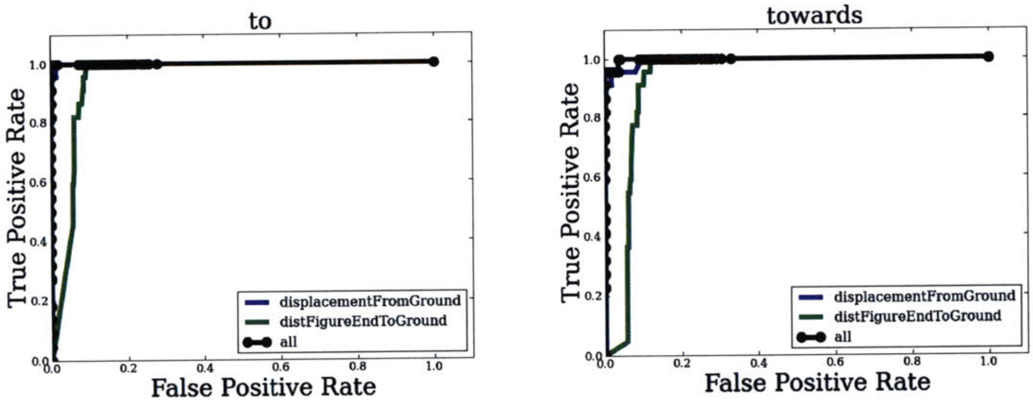


Figure 3-5: Performance of "to" and "towards," overall, and on each feature individually.

more than one feature is necessary to capture the semantics.

Figure 3-5 shows the performance of features for "to" and "towards." The most important feature for both is *displacementFromGround.* The raw distance of the endpoint of the trajectory from the ground works less well, because it captures examples where the figure remains near the ground object for the entire trajectory (such as the example shown in Figure 3-1).

Figure 3-7 shows the performance for "across" and "through." The key feature *ratioFigureToAxes* works well for both. *distAlongGroundBtwnAxes* works better for "across" than "through" because it captures the constraint that the path should bisect the ground, which does not exist for "through."

3.3 Phrase-based Spatial Language Video Retrieval

In order to evaluate the models in the context of phrase-based video retrieval, I collected a corpus of prepositional phrases that annotators used to describe a person's motion in surveillance video. An evaluation of video retrieval with more open-ended, complex spatial language discourse is described in Chapter 6.

Annotators watched a video clip from a surveillance camera installed in the kitchen of a home. I collected data from five different annotators. Annotators were shown

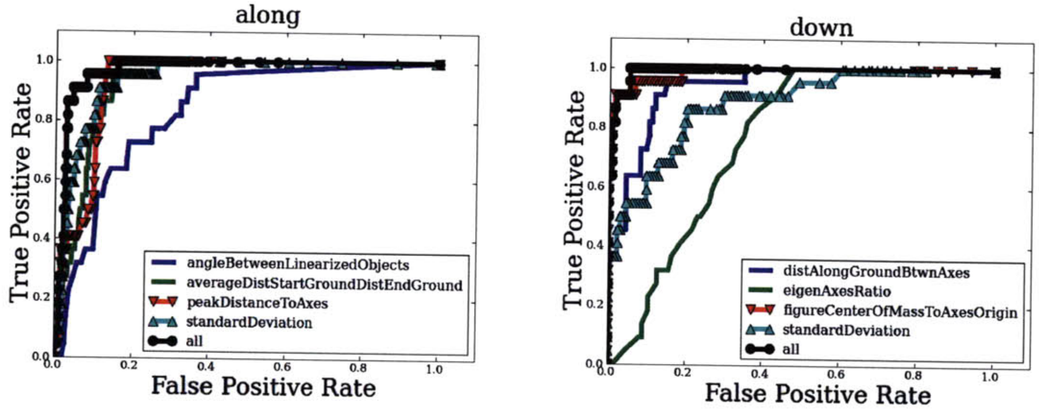


Figure 3-6: Performance of "along" and "down" overall and on each feature individually.

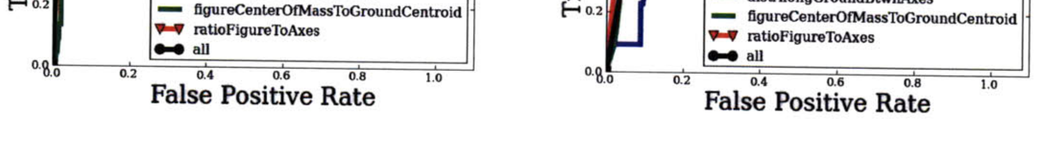


Figure 3-7: Performance of "across" and "through," overall and on each feature individually.

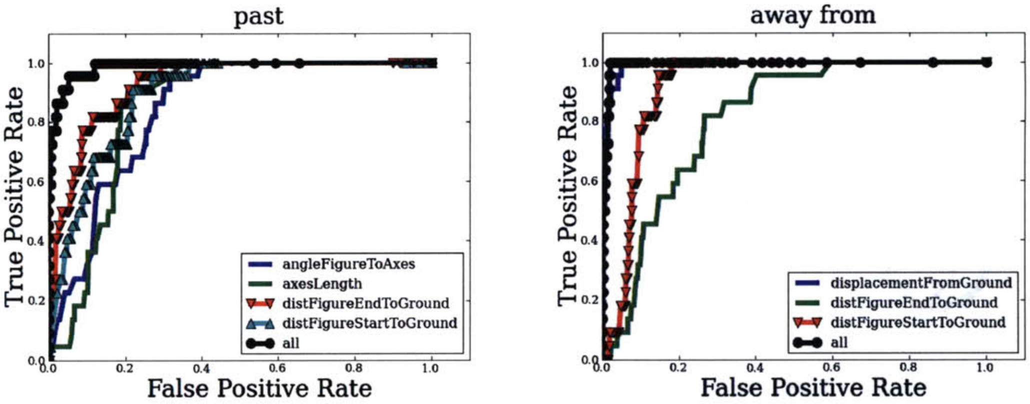


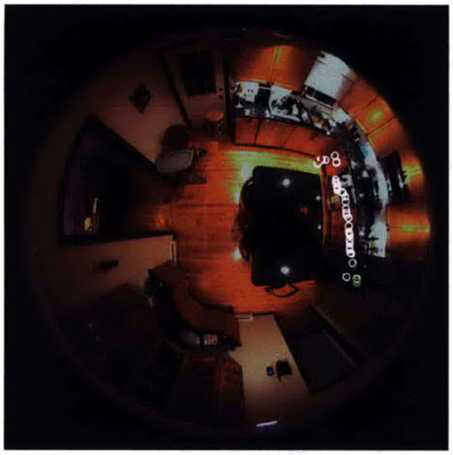
Figure 3-8: Performance of "past" and "away from" overall and on each feature individually.

video clips from two days of video. To focus on prepositions describing movement, annotators saw only tracks that were longer than four seconds, and where the distance between the first and last points in the track was larger than 200 pixels. Clips were shown in random order drawn from two days of data, and each clip appeared in the data set three times in order to collect multiple descriptions from the same annotator for the same clip.

A person's location in the video was marked automatically by a tracker. I used a motion-based tracker that I developed using the SwisTrack free software tracking pipeline [Lochmatter et al., 2008]. When a person moves in the video, the tracker detects the location of the motion, and either creates a new track, or adds the detected point to an existing track. When a person stops moving, the track is ended. These boundaries are often, but not always, reasonable places to start and stop video playback, since they correspond to the start and stop of a person's motion.

The video was overlayed with labels marking the location of non-moving objects such as the refrigerator, doors, and cabinets. Annotators were asked to try to use those labels in their descriptions, but were not required to use them.

After annotation, each clip in the data set had up to fifteen descriptions associated with it, with an average of 10.7. Figure 3-9 shows a frame from a clip in our corpus, together with some descriptions. Figure 3-10 shows a histogram of the frequency of the



to the counter.
along the east side of the island.
from the refrigerator.
to the cabinet.
across the kitchen.

(a) Clip.

(b) Annotations.

Figure 3-9: Annotations for a video clip from the restricted corpus. Annotators completed the sentence "The person is going..." with one prepositional phrase.

descriptions that appeared in the corpus, color coded by annotator, while Figure 3-11 shows the distribution of prepositions in the corpus. From the histograms, it seems that annotators tend to reuse descriptions, rather than inventing an entirely new one for each track. Despite this tendency, a diverse set of spatial prepositions appears in the corpus.

Figure 3-12 shows the distribution of ground objects used in the corpus. Ambiguous ground objects such as "the door" and "the counter," which appeared more than once in the kitchen are resolved through manual annotations. Descriptions which resolved to more than one ground object were excluded from the evaluation. Examples of descriptions rejected for this reason include "from one counter to the other," "back and forth," and "through both doors."

All descriptions that the system successfully parsed and had associated ground objects were included in the evaluation. Table 3.1 shows the number of tracks that annotators skipped, and the number of parse failures for the tracks used in the evaluation.

Corpus Size	
tracks left blank	971
grounding and parsing failures	393
parse successes	7051
total	8415

Table 3.1: The size of the corpus, together with numbers of tracks excluded for various reasons.

Results

I used the restricted corpus to train classifiers for spatial prepositions and evaluate the performance of the classifiers. In order to train classifiers for prepositions in the corpus, each description was converted into a training example. If the description used a preposition, it was treated as a positive training example for that preposition, and if the description used some other preposition, it was treated as a negative training example, following Regier [1992]. For example, the description "around the island" paired with a video clip was treated as a positive example of "around" and a negative example of "to." This heuristic is imperfect: a track that is "along the island" may also be "around the island." In some of these ambiguous cases, I excluded similar spatial prepositions from the training and test sets. For example, for "to," I excluded examples labeled with "towards," "out," "through," and "into" because a track labeled "out the door" was often a good example of "to the door." Data was separated into training and testing by track: all descriptions associated with a track appeared either in the training set or the test set. 80% of the tracks were used as training data, and the rest as test data.

Unique objects that rarely move such as "the kitchen" or "the coffee maker" are resolved to pre-annotated regions in the video. Some noun phrases such as "the door" or "the counter" can not be automatically resolved in this way because there is more than one in the room being recorded. For example, in Figure 3-13, if an annotator labeled the clip "past the counter," the proper ground object is the east counter; on the other hand, if the label was "to the counter," it probably refers to the north

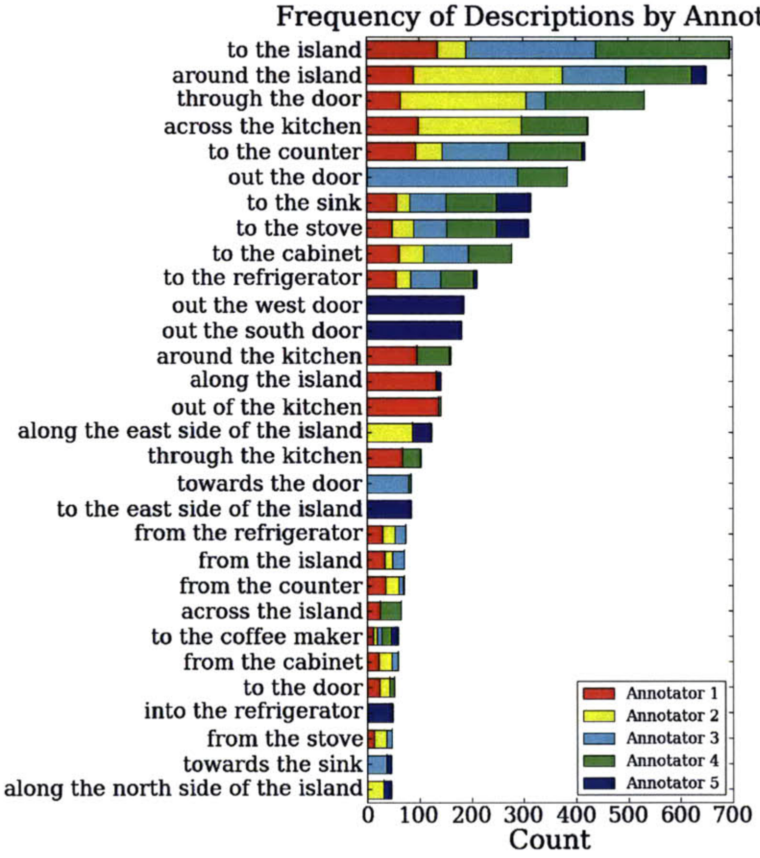


Figure 3-10: Histogram of the frequencies of various descriptions in the restricted video retrieval corpus.

counter. To sidestep this issue, I manually labeled ambiguous objects with the proper ground.

To visualize classifier performance, I report ROC curves. All results use the naive Bayes classifier in the Orange Data Mining library [Demsar and Zupan, 2004]. I measured the performance of a classifier trained using all features, as well as one trained on each feature in isolation, to see how well each feature works on its own. It is possible a classifier would perform even better with a subset of the features. Although the aim is to use models to support video retrieval, this evaluation does not directly measure retrieval performance, but rather the effectiveness of the classifiers

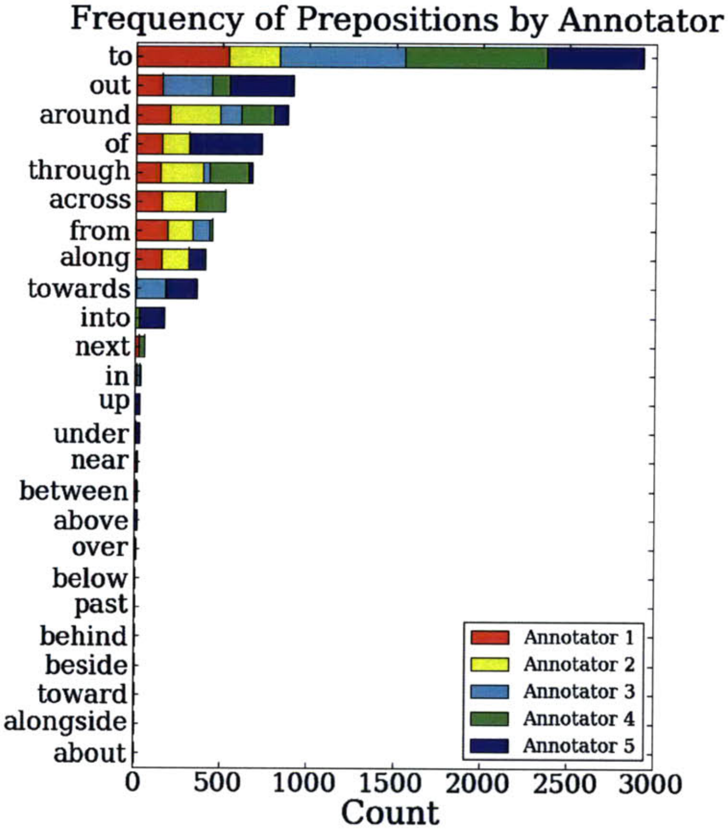


Figure 3-11: Histogram of the prepositions in the corpus.

at capturing the semantics of spatial prepositions that might be used in natural language queries. An evaluation of end-to-end video retrieval performance on open-ended queries is presented in Chapter 6.

Figure 3-14 shows the performance of various binary classifiers for the spatial preposition "to." The classifier trained using all the features clearly performs the best. An alternative interface to search for people going "to the sink" is to manually specify a region of interest, for example by drawing a bounding box. The two features, **numInteriorPoints** and **endpointsInGroundBoundingBox**, capture this heuristic, and perform quite poorly on their own. This result implies that a user searching for people going "to the sink" would be better served by an explicit model of the meaning of "to," implemented in terms of a combination of features, than they

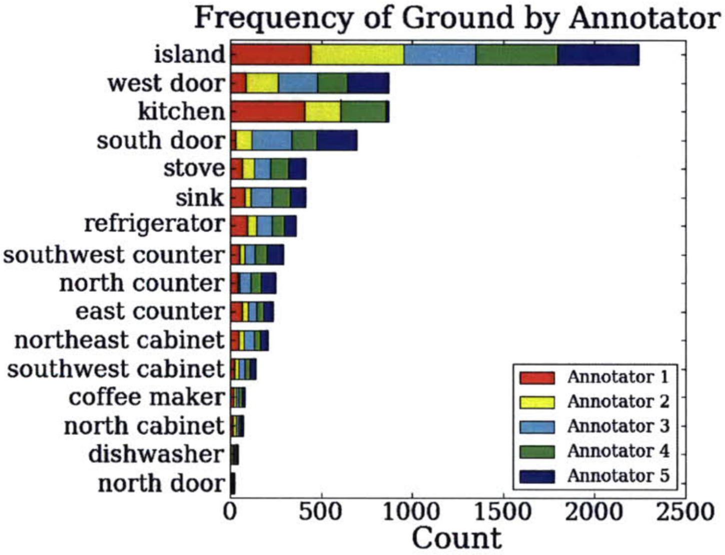


Figure 3-12: Histogram of ground objects used to label tracks in the corpus. Each ground corresponds to a specific object in the camera's visual field; the mapping was determined from human annotations.

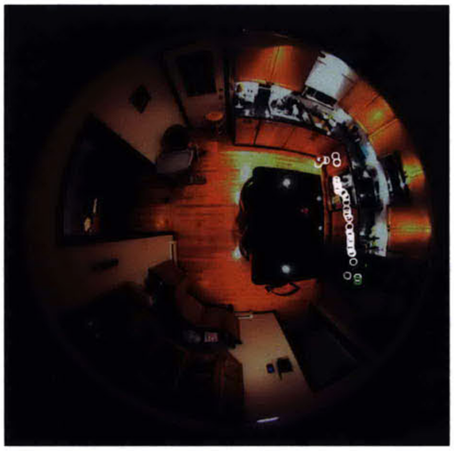


Figure 3-13: A frame from a clip in the corpus. Descriptions for this clip included "to the counter," "along the east side of the island," "from the refrigerator", "to the cabinet," and "across the kitchen."

would be by a query interface in which they drew a bounding box around a region of interest.

In an earlier paper [Tellex and Roy, 2009], I analyzed "across" based on binary

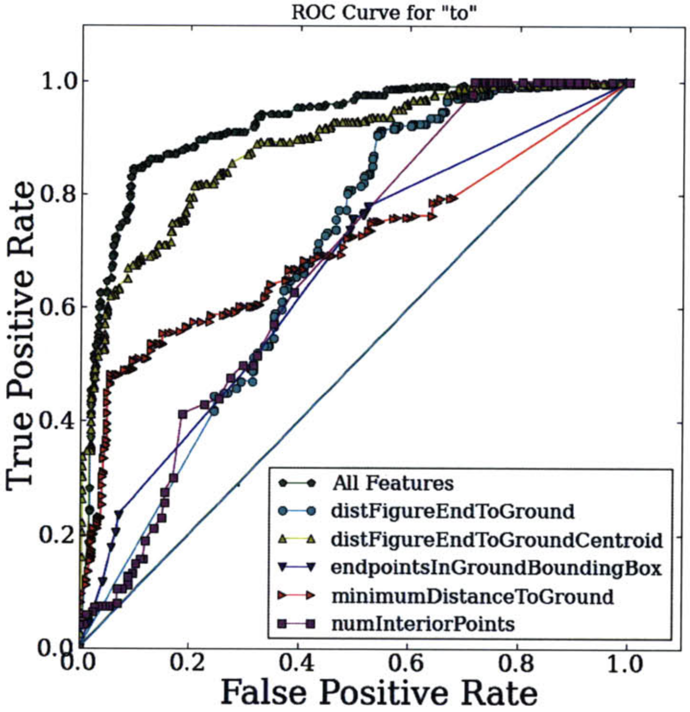


Figure 3-14: Performance of classifiers for "to," with examples containing "out," "through," "towards," and "into" excluded.

annotations, in which annotators marked whether a video clip matched a query such as "across the kitchen." There I found that the feature **ratioFigureToAxes** was critical to good performance, and other features performed poorly on their own. In this corpus, the feature **figureCenterOfMassToGroundCentroid** is also effective on its own. Possibly the difference is due to the different tasks in the two papers: the methodology of collecting natural language descriptions for clips yields fewer borderline "across" examples, changing which features work the best.

"Through" and "out" use a subset of the features used by the "across" classifier. For "out", the feature **ratioFigureToAxes** performed the best. This feature captures the degree to which the figure moves from one point on the boundary of the ground to another point. Both of these spatial prepositions are somewhat problematic in this domain because the tracks do not extend beyond a single camera. When

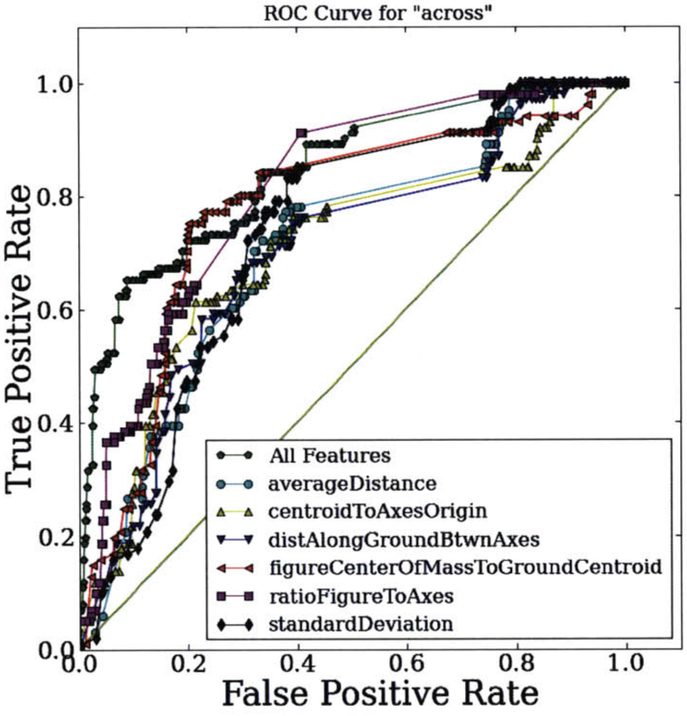


Figure 3-15: Performance of classifiers for "across."

an annotator wrote "through the door," the system saw a track that extended to the door and then stopped. The following chapter will explore the usage of these words in a corpus of natural language directions, leading to a more satisfying representation.

The results for "along" are shown in Figure 3-18. I report the performance of a classifier trained on all features, and on all features except **visibleProportionFigureGround**. This feature is the only feature (so far) which requires additional context from the environment besides the geometry of the figure and ground: it must know about obstacles in the environment which can prevent the figure from being visible from the ground. I added this feature because a classifier for "along" trained on a corpus of explicitly labeled positive and negative examples of "along the right side of the island" sometimes returned tracks that were along the left side of the island. I hoped that adding features that referred to obstacles in the environment would alleviate this problem, but so far have not found an effect.

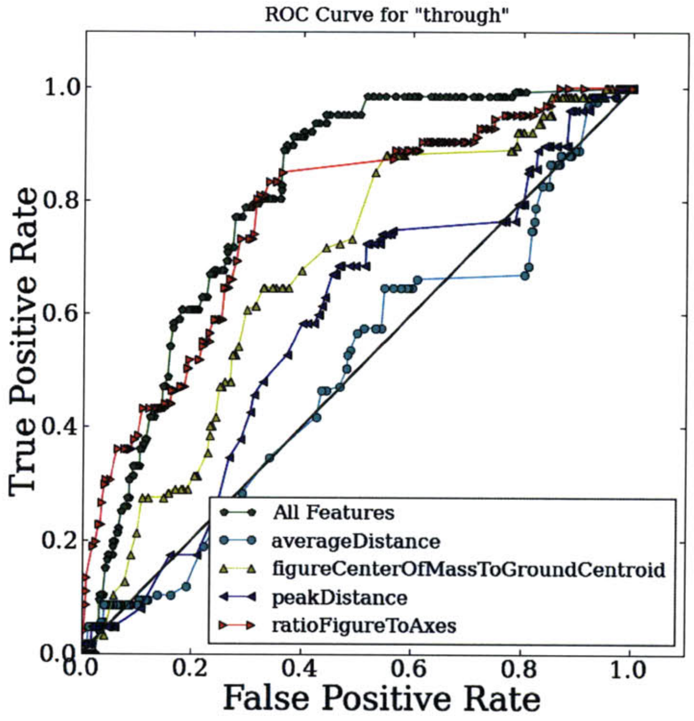


Figure 3-16: Performance of classifiers for "through."

Figure 3-19 shows the performance of the classifier for "around." As it turned out, the most common use of "around" was "around the island," so although the system performs well, it probably does not generalize well to other examples. Interestingly, the feature **distStartToGround** performs much better than **distEndToGround**, despite the bias in the corpus for the spatial preposition "to" compared to "from," and despite evidence that people pay more attention to the goal of a spatial motion event [Regier and Zheng, 2007].

Overall these results are promising. I have identified a set of features that can successfully classify examples in the corpus. The performance on this task is encouraging because it indicates that the features are capturing important aspects of the semantics of spatial prepositions. A major remaining issue is that in a retrieval context, even a low false positive rate can yield poor retrieval performance if there are many more negative examples in the corpus than positive examples. Despite this issue, models

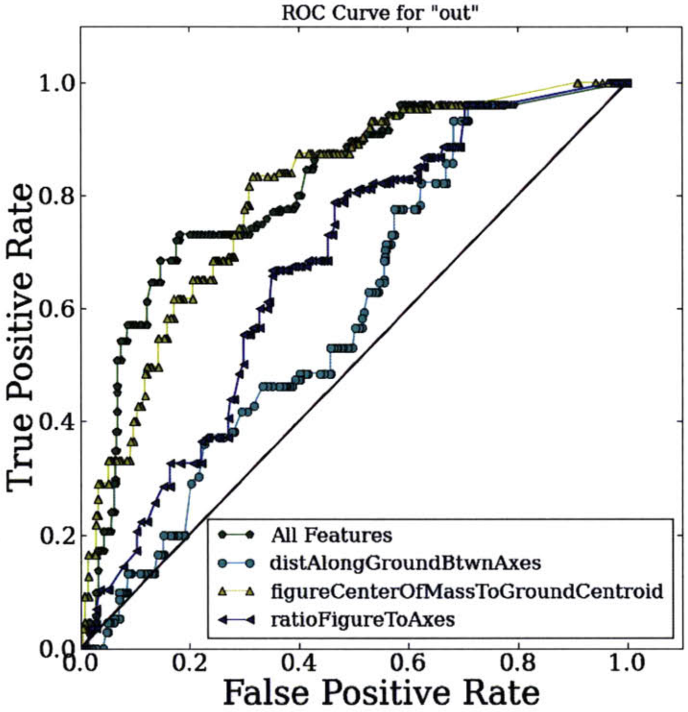


Figure 3-17: Performance of classifiers for "out," with examples containing "towards," "through," and "to" excluded.

for the meanings of spatial prepositions are a promising path forward to a natural language video retrieval. Chapter 6 describes an end-to-end retrieval evaluation of a model for understanding not just phrases but spatial language discourse.

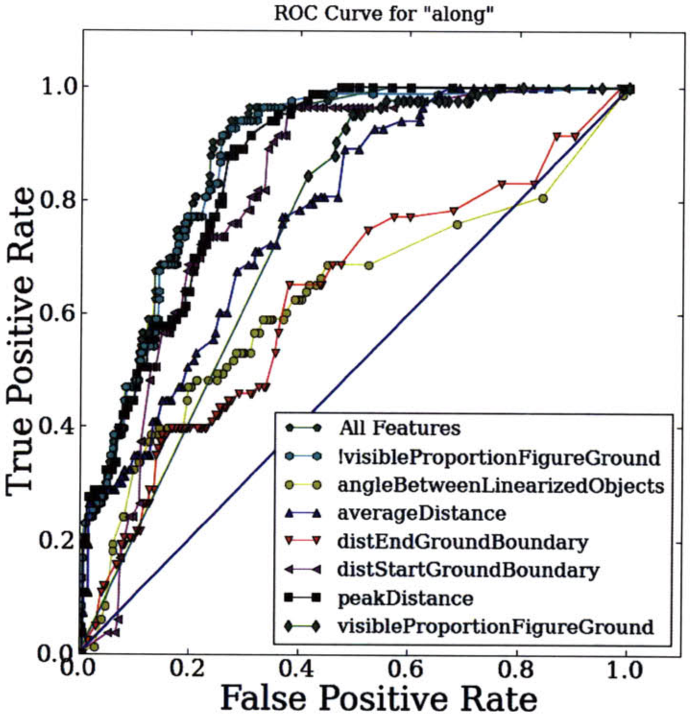


Figure 3-18: Performance of classifiers for "along." **!visibleProportionFigureGround** is a classifier trained on all features except **visibleProportionFigureGround**.

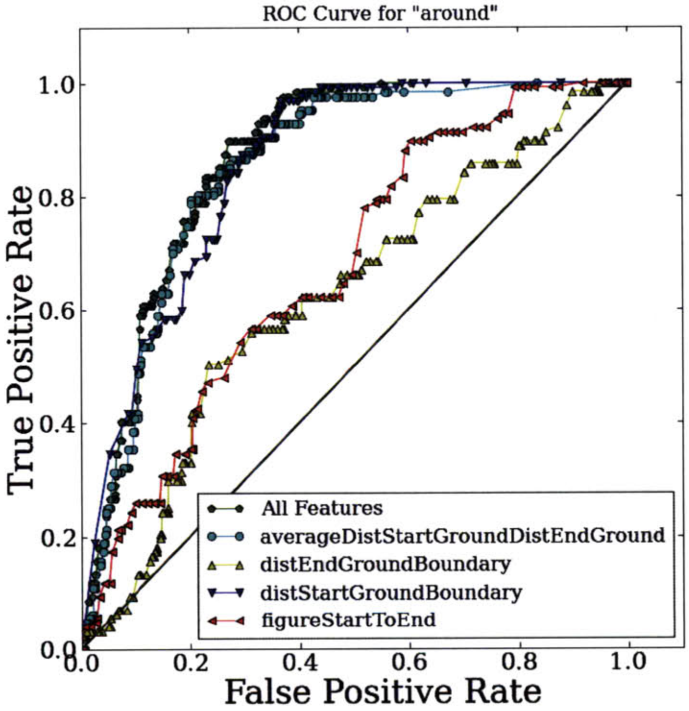


Figure 3-19: Performance of classifiers for "around."

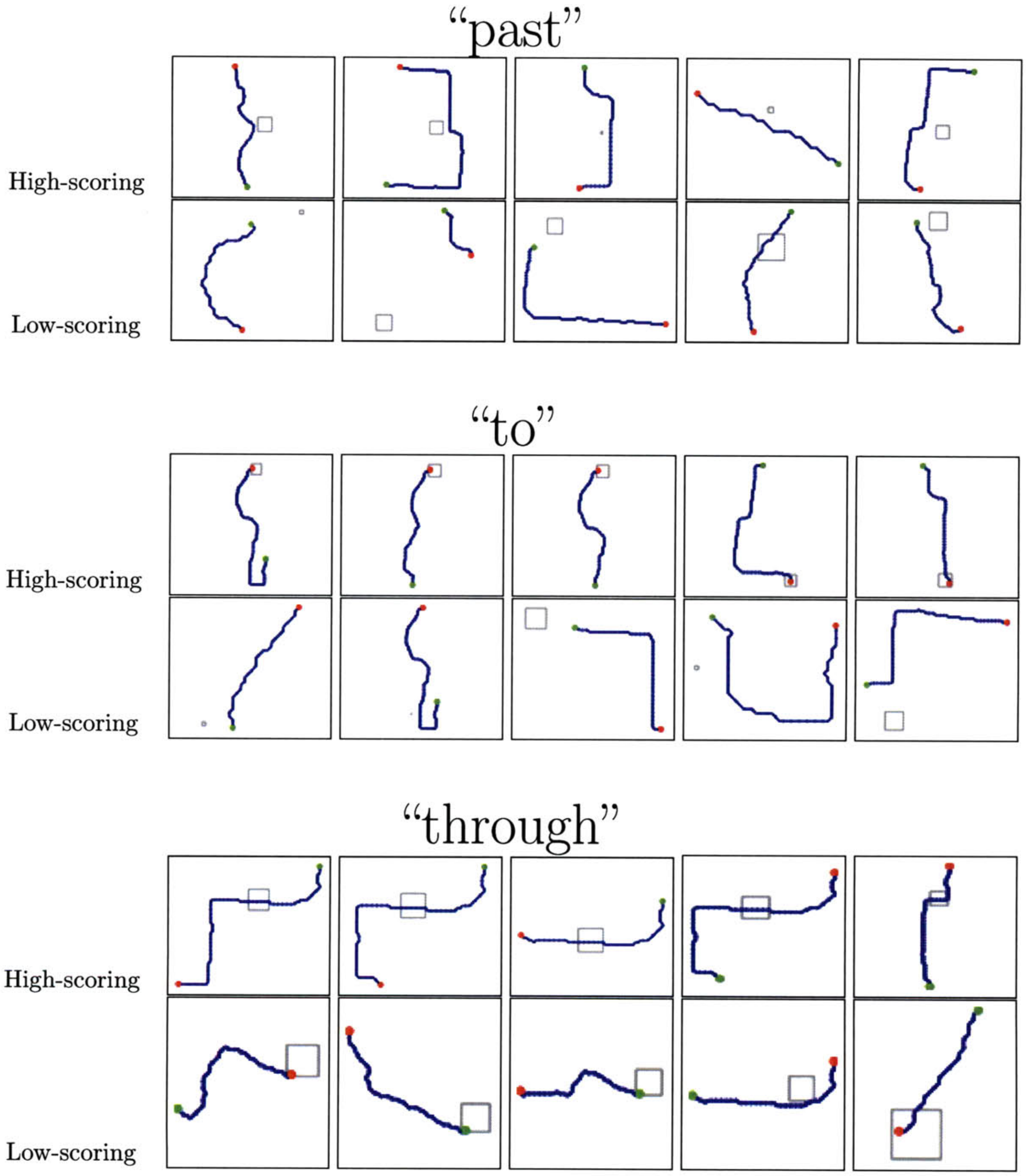


Figure 3-20: Five high scoring and five low scoring examples that were found in the direction understanding topological map for several spatial prepositions.

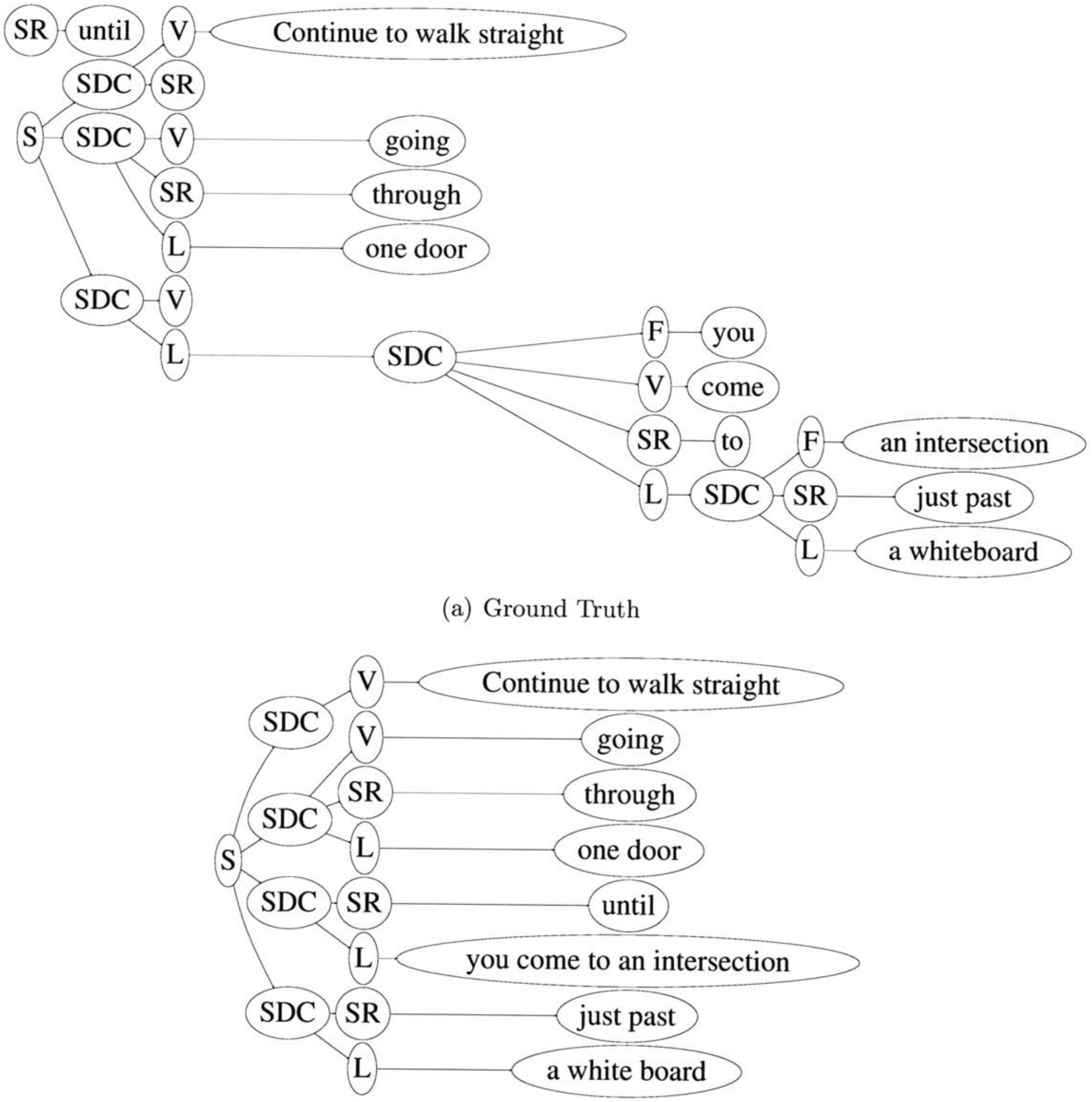
Chapter 4

Spatial Language Discourse

Spatial language created by untrained users is challenging, containing rich syntactic structure and unconstrained vocabulary. In order to robustly understand this type of language, I introduce a novel a semantic structure extracted from the language called a *spatial description clause* or SDC for short, developed jointly with Thomas Kollar. SDCs enable the system to process language in the various corpora used as part of this research. The following chapter describes a probabilistic model based on SDCs that finds paths in the environment corresponding to the natural language input.

4.1 Spatial Description Clause (SDC)

Each spatial description clause (SDC) consists of a *figure* (the subject of the sentence), a *verb* (an action to take), a *landmark* (an object in the environment), and a *spatial relation* (a geometric relation between the landmark and the figure). Any of these fields can be unlexicalized and therefore only specified implicitly. For example, in the sentence "Go down the hallway," the figure is an implicit "you," the verb is "go," the spatial relation is "down," and the landmark is "the hallway." SDCs are hierarchical. For the sentence "Go through the set of double doors by the red couches," the top level SDC has a verb, "go," a spatial relation, "through," and a landmark, "the set of double doors by the red couches," while the landmark contains a nested SDC with figure "the set of double doors," spatial relation "by" and landmark "the red



(a) Ground Truth

(b) Automatic

Figure 4-1: Ground-truth and automatically extracted SDCs for the sentence, "Continue to walk straight, going through one door until you come to an intersection just past a white board." Here, *S* is the entire sentence, *SDC* is a spatial description clause, *F* is the figure, *V* is the verb, *SR* is the spatial relation, and *L* is a landmark.

couches." Figure 4-1(a) shows the hierarchy of SDCs for a sentence in the route instruction corpus.

I hand-annotated the text of 300 directions in a corpus of natural language route instructions through indoor environments in order to verify that SDCs are capable of capturing the linguistically expressed structure of the natural language [Kollar et al.,

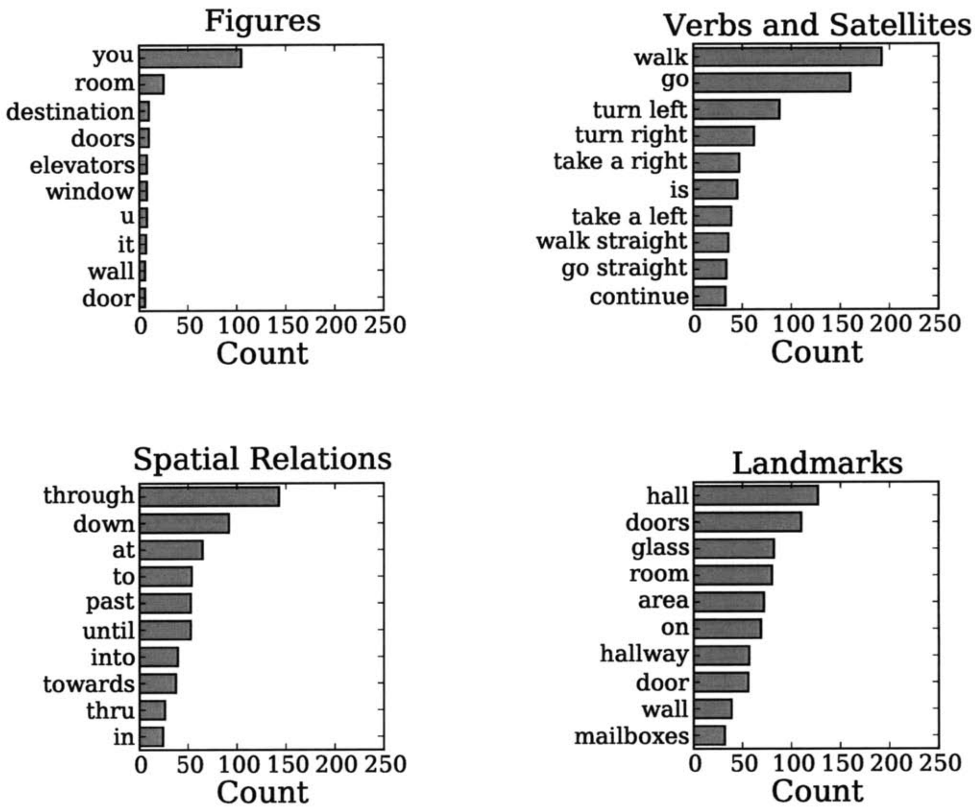


Figure 4-2: Histogram showing the most frequent words that appear in each of the fields of an SDC from the route instruction. For Figures and Landmarks, similar phrases have been grouped together.

2010]. The corpus was collected by Nicholas Roy and his students, and it is described more completely in Chapter 5. Nearly all of the sentences in the dataset can be parsed into SDCs that correctly preserve the semantics of each word in the sentence, with very few (7.29%) orphaned words. Virtually all of the orphaned words are stop words. Figure 4-2 shows the top ten words that appeared in each field of an SDC in the route instruction corpus.

Several types of sentences could not be annotated within the SDC framework. First, multi-argument verbs do not fit in the framework, as in the sentence "Follow the atrium all the way to the right." Here both "the atrium" and "all the way to the right" are arguments to verb "follow." Second, some phrases such as "in order" do not correspond to any of the slots in the SDC, as in "Move past the classrooms 144, 141, and 123 in order." Third it does not capture disjunction, such as the descriptive sentence "You must turn right or left." Fourth, it does not represent negation as in "Do not go through the glass and wooden doors." Finally, the framework does not explicitly represent ambiguous attachment in sentences such as "Go down the hall to the right of the whiteboard." This sentence could be about "the hall to the right of the whiteboard" or alternately could mean "Go... to the right of the whiteboard." Both these meanings can be represented as two different SDCs, but the fact that both are possible is not directly represented. These limitations are part of a long tail of linguistic phenomena that sometimes occur in natural language directions, and require a more complex framework. SDCs capture important parts of the semantics of route instructions in the corpus, and are efficient to extract and use in inference.

4.2 Automatically Extracting SDCs

Although SDCs are hierarchical, like the hierarchical structure of natural language, inference is simplified if they are modeled sequentially. Section 4.3 describes how to perform inference for a sequence of SDCs; a full hierarchical model remains future work.

To automatically extract a sequence of SDCs from natural language text, the system uses a conditional random field (CRF) model [Kudo, 2009]. The CRF takes as input a natural language sentence and labels each word in each sentence with one of the four possible fields (*figure*, *verb*, *spatial relation* and *landmark*), or none. The model was trained using an annotated corpus of natural language route instructions. A greedy algorithm groups continuous chunks together into SDCs. Figure 4-1(b) shows the SDCs generated by this component for one sentence. Although it lacks the hierarchical structure of the annotated data (as in Figure 4-1(a)), the SDCs capture the sequential structure of the directions, and segments the key components of each phrase. Quantitatively, 60% of the SDCs produced by the CRF correspond exactly to an SDC in the hand-annotated ground truth. To measure inter-annotator agreement, a second person annotated the SDCs in the corpus, and also had 60% agreement with my annotations. Figure 4-3 shows the performance of the system against ground truth for exact matches, and for each subfield of the SDC. When the automatic algorithm makes a mistake, it is usually a boundary problem, for example including the spatial relation and landmark, but excluding the verb. In these cases, the annotations still contain structured information that can be used to reason based on the natural language input.

I used the same SDC extractor for the open-ended route instructions corpus. The most common error it makes in this corpus is ambiguous prepositional phrase attachment. For instance, in the example "Go down the hall to the bathroom," the extractor creates one SDC (V: "Go", SR: "down", L: "the hall to the bathroom") instead of two. [(V: "Go", SR:"down", L:"the hall"), (SR: "to", L:"the bathroom")] However, overall it appears to work quite robustly.

SDCs provide the structure for decomposing spatial language into components. Each component can be modeled separately and composed back together using independence assumptions defined by the SDC. This decomposition enables separate models to be learned for the meanings of verbs, spatial relations, and landmarks. By decomposing models in this way, much less training data is required to learn the models, while a large variety of language can be understood. Because it does not rely on a full correct parse, the system is more robust to sentence fragments and ungrammatical language.

4.3 Modeling Spatial Language with SDCs

Spatial description clauses (SDCs) described in the previous chapter provide a conceptual framework that the system uses to understand language. However, to actually map between language and the world requires more machinery. This chapter describes a probabilistic model developed jointly with Thomas Kollar that maps between language and the world. The factors in this model are based upon the structure of the spatial description clause, and incorporate the distribution for spatial relations de-

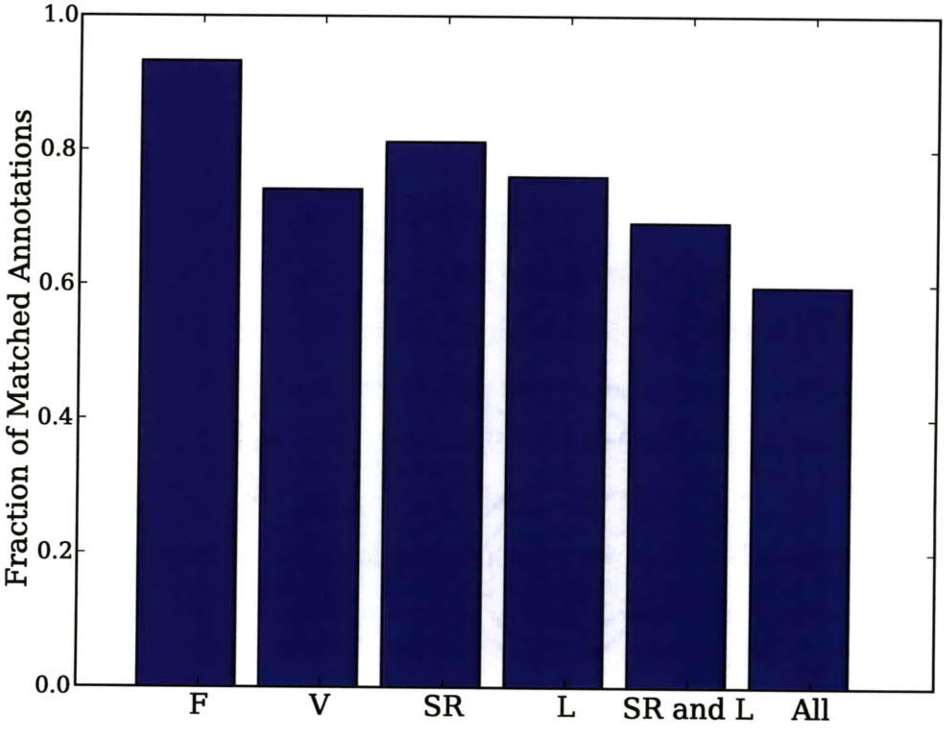


Figure 4-3: Performance of the model that extracts spatial description clauses from text on each component of the clause alone, and overall. The model was trained on directions from floor 1, and tested on directions from floor 8.

scribed in the previous chapter. The following chapters describe how this model was actually used to understand language from the two application domains. Language is mapped to contextual information in the world via a probabilistic model of the joint distribution of language and contextual information:

$$p(language, context) \tag{4.1}$$

The structure and independence assumptions of this model correspond to the assumption that spatial language corresponds to a sequence of SDCs, as described in the previous chapter. To compute this distribution, the first step is to rewrite it in terms of the SDCs extracted from the query, and represent the context as a trajectory t.

$$p(language, context) = p(SDC_1 \dots SDC_N, t) \tag{4.2}$$

The general inference problem is to find a trajectory t that maximizes this distribution:

$$\arg\max_t p(SDC_1 \dots SDC_N, t) \tag{4.3}$$

Assuming SDCs are independent of each other yields the following factorization:

$$p(SDC_1 \dots SDC_N, t) = \prod_i p(SDC_i, t) \tag{4.4}$$

In general, an SDC_i may apply to only part of a trajectory t, especially for longer trajectories. This alignment problem is approximated using a topological map for direction understanding, and by assuming that each SDC applies to the entire trajectory for video retrieval.

For natural language direction understanding and video trajectory search, the goal is to find a trajectory that maximizes this distribution. The system does not know which physical object in the world o is associated with a particular SDC.

The model assumes the spatial relation and landmark fields of the SDC are referring to a particular object in the environment. However, since the correct object is unknown, the model marginalizes over all the possibilities, O.

$$p(SDC, t) = \sum_{O} p(SDC, t, o) \tag{4.5}$$

The inner term can be rewritten in terms of the fields of the SDC, figure f, verb v, spatial relation s and landmark l and then factored:

$$\begin{aligned} p(SDC, t, o) &= p(f, v, sr, l, t, o) & (4.6)\\ &= p(f|t, o)p(v|t, o)p(sr|t, o)p(l|t, o)p(t, o) & (4.7) \end{aligned}$$

This factorization assumes that the fields of the SDC are independent of each other and depend only on the ground object and the trajectory. Because of these independence assumptions, it is possible to model each factor separately, which requires much less training data than learning a model for the joint directly and yields models that can be reused in other contexts.

The core of this thesis is models for the meanings of spatial relations $p(sr|t, o)$ and spatial motion verbs, $p(v|t, o)$. The remainder of this chapter describes how other terms are modeled. $p(f|t, o)$ is the probability that a phrase like "She" maps to a trajectory for a description such as "she walked down the hall." It could be learned based on features in the video, although as yet I have not built models for it.

$p(t, o)$ is a prior based on trajectories and objects. This term is factored:

$$p(t, o) = p(t|o)p(o) \tag{4.8}$$

$p(o)$ is the prior probability on objects and does not affect the inference because it is constant. $p(t|o)$ is the prior probability of paths given observations of the environment, before processing any directions. For example, paths that go down hallways and pass by many doors might be more likely than a path that goes into an office. This distribution could be learned from an appropriate corpus. Instead I assume that paths are independent of objects in the environment in order to focus on incorporating information from the language directions into the model.

4.3.1 Landmarks

In order to ground landmark objects, the system needs to estimate the probability that a landmark noun phrase l such as "the couches in the living room" could be used to describe a concrete object o with a particular geometry and location in the environment, given a trajectory t. One wants to estimate:

$$p(l|t,o) = p(l|o)p(o|t) \tag{4.9}$$

$p(o|t)$ is a mask based on whether a particular object is visible from the trajectory. I assume that objects not visible from a particular trajectory are never used as landmark objects in a description. This assumption saves computation during the inference as all known objects do not need to be considered. A more informed prior would take into account object saliency: how salient a given object is for a particular trajectory.

To model $p(l|o)$, the system extracts nouns, adjectives, and verbs from the landmark phrase, representing the landmark phrase l as a set of words w_i.

$$p(l|o) =p(w_1 \dots w_M|o) \tag{4.10}$$

For example, the words extracted from "the couches in the living room" are "couch," "living," and "room." This factorization assumes that one of the keywords is referring to the physical landmark object, and the other keywords are descriptors or reference objects. For "the couches in the living room," the grounded keyword is couches; this word directly grounds out as the object being referred to by the landmark phrase, and other extracted words are modifiers. However, the system does not know which keyword is the root and which are modifiers. To address this problem the system represents which keyword k is the root with an assignment variable

$\phi \in 1 \ldots M$ which selects one of the words from the landmark phrase as the root and marginalize over possible assignments:

$$p(k_1 \ldots k_M | o) = \sum_{\phi} p(k_1 \ldots k_M, \phi | o) \tag{4.11}$$

Expanding the inner term gives:

$$\begin{aligned} p(k_1 \ldots k_M, \phi | o) &= p(k_1 \ldots k_M | \phi, o) p(\phi | o) && (4.12) \\ &= \prod_i p(k_i | \phi, o) p(\phi) && (4.13) \end{aligned}$$

The prior on ϕ is independent of the physical object o and depends on grammatical features of the landmark phrase; it is modeled as uniform over all the keywords, although a more informed prior would use parse-based features to identify the head noun. The likelihood term can be broken down further depending on the value of ϕ:

$$p(k_i | \phi, o) = \begin{cases} p(k_i \text{ is } o) & \text{if } \phi = i \\ p(k_i \text{ can see } o) & \text{if } \phi \neq i \end{cases} \tag{4.14}$$

$p(k_i$ is $o)$ could be estimated using hypernym/hyponym features from Wordnet and part of speech information, but here it is based on whether k_i matches the label for o in the semantic map of any object in the environment; if it does, it is modeled based on whether o has that tag. Otherwise, it backs off to co-occurrence statistics learned from tags for over a million images downloaded from the Flickr website [Kollar and Roy, 2009]. For example, using this corpus, the system can infer which bedroom is "the baby's bedroom" without an explicit label, since only that room contains a crib and a changing table. This construction allows the system to handle expressions such as "the hallway door" and "the room with the fireplace." This model uses visibility information as a proxy for detailed spatial-semantic models of relations such as "in front of" and "in," and seems to work better than a baseline that uses $\prod_{k_i} p(k_i$ can see $o)$ to approximate $p(k_1 \ldots k_M | o)$.

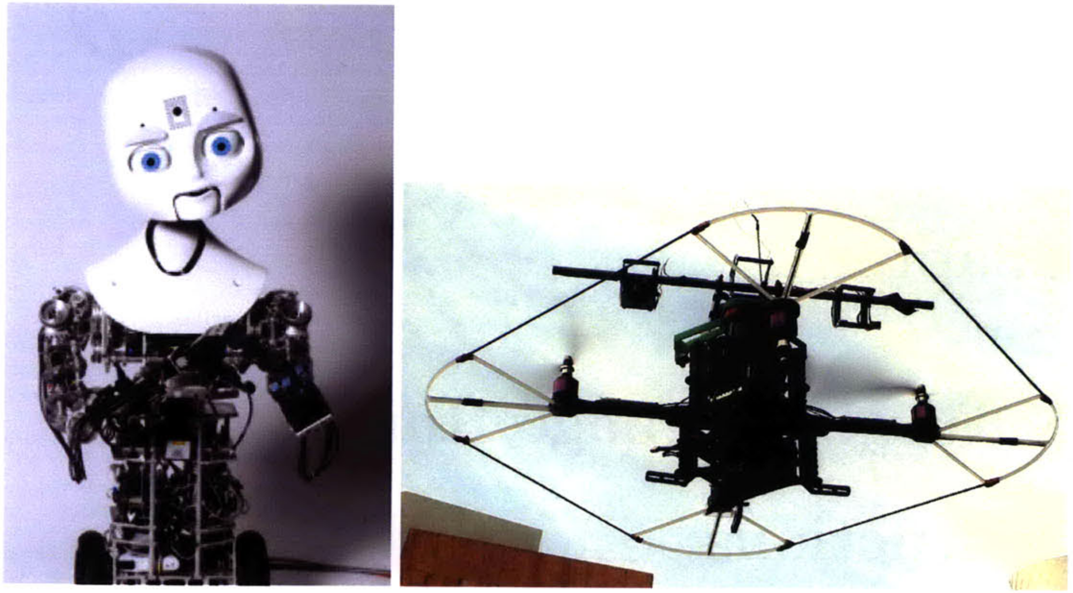
Chapter 5

Natural Language Route Instructions

One of the two application domains in this thesis is that of natural language route instructions for robots. Together with video retrieval, these two domains comprise two key uses of language: obeying commands, and recognizing events based on a description.

Natural language is an intuitive and flexible modality for human-robot interaction. A robot designed to interact naturally with humans must be able to understand instructions without requiring the person to speak in any special way. Understanding language from an untrained user is challenging because the human is not asked to adapt to the limitations of the system, i.e., to limit their instructions to a small vocabulary or grammar. Rather, one wants a system that understands naturalistic language directly as produced by people.

This work is directed toward understanding naturalistic language as part of a larger multi-university effort to develop autonomous robot teammates that collaborate naturally and effectively with humans in a civilian response after a mass casualty event. Two types of robots take part in the scenario: quadrotor helicopters and the Mobile Dexterous Social robot (MDS), an expressive, mobile humanoid (Figure 5-1). In the first-responder scenario, robots will need to interact with both trained and untrained humans to aid the rescue effort. For example, a robot might engage in the



(a) humanoid

(b) helicopter

Figure 5-1: Robot platforms used as part of a disaster-recovery scenario.

following dialog when it finds a human victim:

- **Robot** Someone is on the way to get you out of here. Are there any other people around who need help?
- **Victim** I saw someone in the main lobby.
- **Robot** Where is the main lobby?
- **Person** Exit this room and turn right. Go down the hallway past the elevators. The lobby is straight ahead.
- **Robot** Understood.

The system can infer paths through any environment based on natural language directions. If the robot has explored the entire area *a priori* and has access to a map of the environment, *global* inference searches through all possible paths to find the global maximum of the joint distribution. When a full map is unavailable, the robot uses a greedy *local* inference algorithm that searches for paths using only local information. The system performs global inference using a Viterbi-style algorithm [Viterbi, 1967]

With your back to the windows, walk straight through the door near the elevators. Continue to walk straight, going through one door until you come to an intersection just past a white board. Turn left, turn right, and enter the second door on your right (sign says "Administrative Assistant").

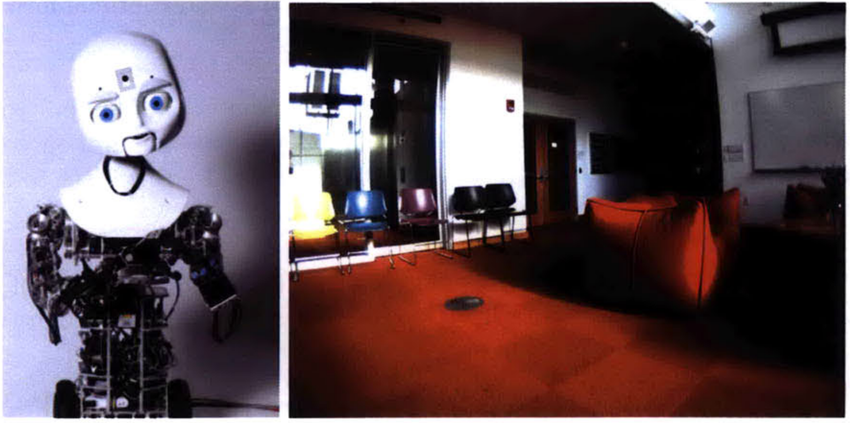


Figure 5-2: A robot such as Nexi (pictured above) takes as input a set of natural language directions and must find its way through a naturalistic environment such as the one pictured.

that finds the most probable sequence of viewpoints corresponding to a given sequence of SDCs. The algorithm takes as input a starting viewpoint, a map of the environment with some labeled objects, and the sequence of SDCs extracted from the directions. It outputs a series of viewpoints through the environment, using the model described above to compute the probability of a transition between two viewpoints.

The local inference algorithm iterates over the SDCs and at each step chooses the next viewpoint v_{i+1} that maximizes $p(sdc_i|v_{i+1}, v_i, o_1 \ldots o_K) \times p(sdc_{i+1}|v_{i+1}, v_{i+2}, o_1 \ldots o_K) \times p(v_{i+1}|v_i) \times p(v_{i+2}|v_{i+1})$. In other words, it looks ahead two SDCS, and chooses the best transition from among the children and grandchildren of the current node. Global inference performs better because it searches through all possible paths to find the one that best matches the descriptions. However, the local inference is more practical for a real robot, because it does not require the robot to have built a complete map of the environment and objects in it before following directions.

With your back to the glass entryways, walk toward the question mark sign. You will continue roughly in this direction as far as you can go, passing: two large white columns, two smaller grey pillars, under two skylights, until you reach a hanging concrete staircase. Continue under this staircase, then turn right at the doors and continue forward until you see elevators. Stop here.

Walk straight for a while. You should pass a computer station on your left, and then the classroom ’144’ on your right. Keep going as you pass a mural on your right then the classroom ’124’ on your right. Keep straight with a children’s’ center to the left. Take a left turn until you get to the elevators on your left.

Walk down the hall to the right of the room. Continue walking straight. Go to the right of the room labeled MIT Libraries. Continue to walk straight through the halls. The building will curve and you will walk around a staircase. There is a radar exhibit in your final location.

Walk toward the large white question mark and pass it such that it’s on the left. Keep walking in that direction passing a small “MIT Libraries” room on the left and wooden trash / recycling bins on the right. You’ll come to an open hallway with bulletin boards and big chalk boards. Walk past those until you reach concrete stairs overhead. Walk under the stairs, then turn right. Head toward the glass doors. Stop at the elevators on the left.

Figure 5-3: Example directions from the route instruction corpus.

5.1 Corpus

I used a corpus of natural language route instructions collected by Thomas Kollar, Emma Brunskill, Sachi Hemachandra, and Nick Roy [Kollar et al., 2010]. Figure 5-3 shows sample directions from this corpus. Directions were collected from an office environment in two adjoining buildings at MIT. The goal in collecting this corpus was to obtain examples of natural language directions produced by a human for another human to develop and evaluate a system for automatically following directions. Fifteen subjects wrote directions between 10 different starting and ending locations, for a total of 150 directions. Subjects were solicited by placing flyers around MIT and were selected for inclusion in the study if they were between the ages of 18 and 30 years old, were proficient in English, and were unfamiliar with the test environment. The pool was made up of 47% female and 53% male subjects from the MIT community, primarily students and administrators.

When collecting directions, subjects were given a tour of the building to familiarize

Table 5.1: The performance of the models at 10 meters.

Algorithm	% correct Max Prob	 Best Path
Global inference w/ spatial relations	48.0%	59.3%
Global inference w/o spatial relations	48.0%	54.7%
Local inference w/ spatial relations	28.0%	42.0%
Local inference w/o spatial relations	26.7%	30.7%
Wei et al. [2009]	34.0%	34.0%
Last SDC only	23.0%	24.0%
Random	0.0%	–

them with the environment. Then they were asked to write down directions from one location in the space to another, as if they were directing a friend. Subjects were allowed to wander around the floor as they wrote the directions and were not told that this data was for a robotics research experiment. Experimenters did not refer to any regions in the environment by name, instead using codes labeled on a map.

In order to enable an offline evaluation, the corpus includes a log of a robot's observations of the environment. To collect the dataset, a mobile robot instrumented with a LIDAR and camera drove through the environment. The log thus contains odometry, laser scans, and camera images. The laser scans and odometry were used by a SLAM module in order to create a map as the robot explored the environment [Grisetti et al., 2007]. The log enables the testing of models for following directions from the corpus offline, without the overhead of deploying a physical robot in a user study.

5.2 Results

To evaluate the technical feasibility of the approach, I performed a component-level evaluation of the system, measuring its performance at following natural language directions from the corpus. For each set of directions, the system tried all four possible starting orientations. Two evaluation metrics were used. For the maximum probability metric, only the highest probability path from the four starting orientations is evaluated. In the best-path metric, only the path that ended up closest to the true destination is evaluated. For the latter metric, the true starting orientation of the subject at the beginning of each set of directions was difficult to automatically determine. Figure 5-4 shows a comparison of the model to three baselines: on the horizontal axis is the distance from the final location of the inferred path to the correct destination, on the vertical axis is the percentage correct at that distance. Performance differences at 10 meters are shown in Table 5.1. I present performance at a threshold qualitatively close to the final destination in order to compare to human performance on this dataset, which is 85%.

The first baseline (*Random*) is the expected distance between the true destination and a randomly selected viewpoint. The second (*Last SDC*) returns the location that best matches the last SDC in the directions. The third baseline (*Landmarks Only*) corresponds to the method described by Wei et al. [2009], which performs global inference using landmarks visible from any orientation in a region, and no spatial relations or verbs. The global inference model significantly outperforms these baselines, while the local inference model slightly outperforms Wei et al. [2009] despite not performing global search.

The performance of the model with and without spatial relations is of particular interest, since spatial relations are a key difference between this model and previous work. Figure 5-5 shows the performance of these models for all subjects, while Figure 5-6 shows the performance for the subject whose directions had the highest performance with the global inference algorithm with spatial relations. Spatial relations do not contribute much to the performance of the global inference algorithm, but do increase the performance of the local inference algorithm. For one subject, they raise the performance of the local search algorithm into the range of the global inference algorithms. Possibly this effect is because when deciding to go through a particular door, the global inference algorithm searches on the other side of that door, and if landmarks farther along in the sequence of SDCs match that path, then it will go through the door anyway. In contrast, the local search approach benefits more from spatial relations because it cannot see the other side of the door, so relying on the geometric features of the path helps to disambiguate where it should go. Fig-

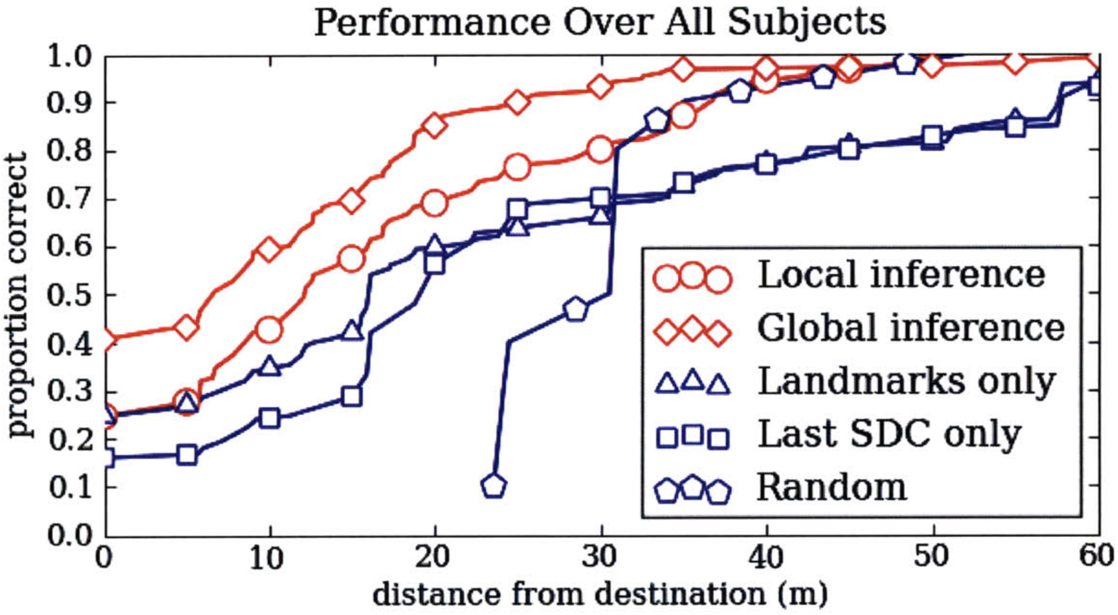


Figure 5-4: Comparison of our model to baselines. Versions of the model are shown in red; baselines are in blue.

ure 5-7(a) and Figure 5-7(b) show the paths inferred by the two models for the set of directions. Without spatial relations, the model is content to stay in the first room, from which it can see objects, such as a whiteboard and a door, that occur in the directions. In contrast, the model that uses spatial-relations goes through the first door when told to "walk straight through the door near the elevators" and ends up at the correct final destination. This result suggests that the role of spatial relations in natural language directions is to help the direction follower disambiguate these local decisions.

The most significant improvement in performance over the system corresponding to Wei et al. [2009] comes from the model of verbs with viewpoints, suggesting that the combination of verbs and landmarks is critical for understanding natural language directions. It is surprising that a relatively simple model of verbs, involving only left, right, and straight, caused such a large improvement compared to the effect of spatial relations.

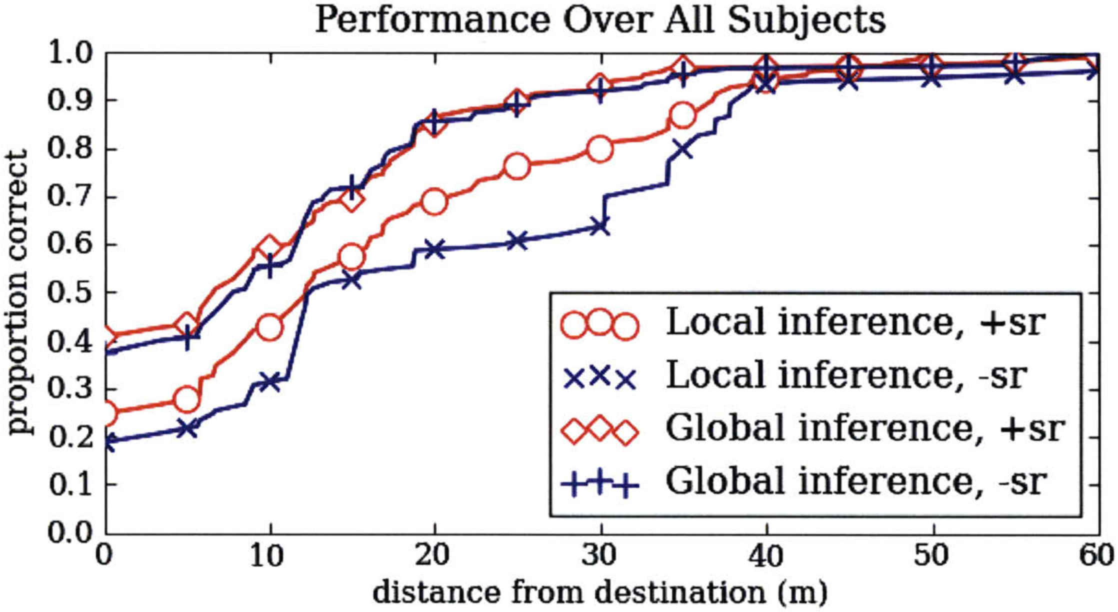


Figure 5-5: Comparison of global inference algorithm with local inference, with and without spatial relations, using the best-path metric.

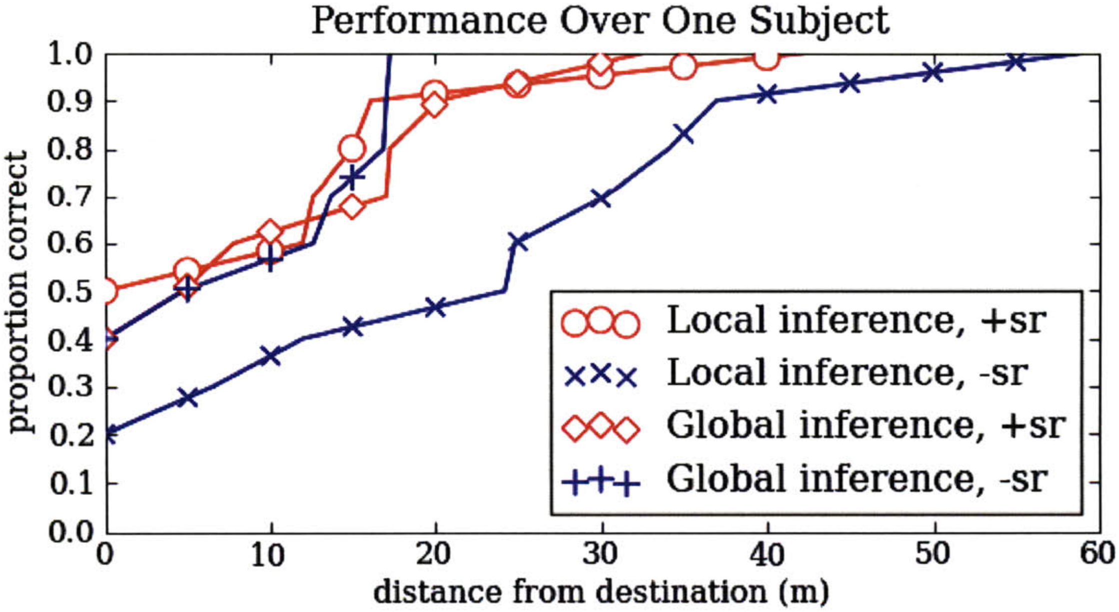
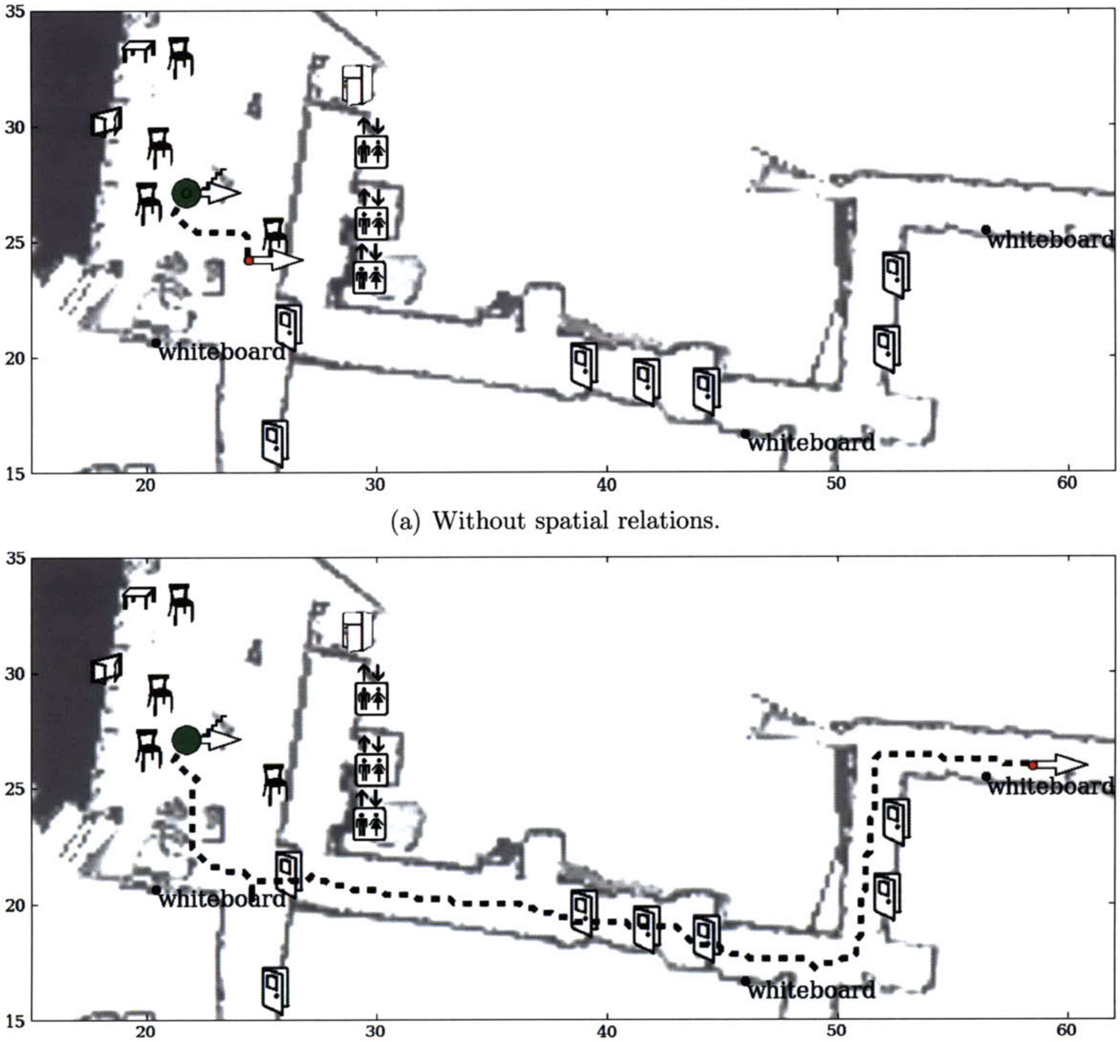


Figure 5-6: Comparison of the global inference algorithm with the greedy approach, with and without spatial relations for directions from one of the best-performing subjects, using the best-path evaluation metric.



(a) Without spatial relations.

(b) With spatial relations.

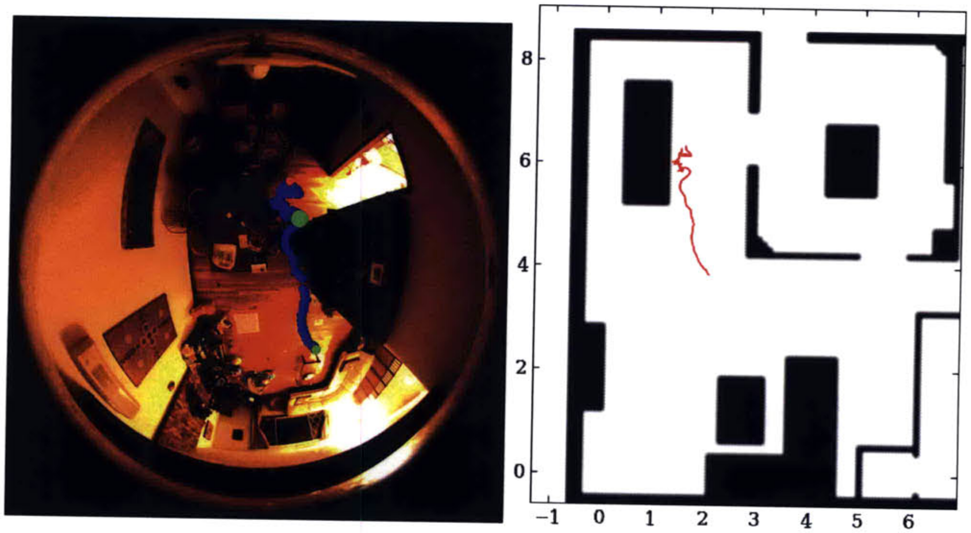
Figure 5-7: Paths output by the local inference system with and without spatial relations for the instructions: "With your back to the windows, walk straight through the door near the elevators. Continue to walk straight, going through one door until you come to an intersection just past a white board. Turn left, turn right, and enter the second door on your right (sign says "Administrative Assistant")." The path without spatial relations is extremely short and completely incorrect.

Chapter 6

Video Retrieval

Spatial language video search is the problem of finding a video clip that matches a natural language description, such as "show me people going across the kitchen." Searching video with natural language has many applications in military, health care, and science. In this thesis the video corpus consists of data recorded from a fish-eye camera installed in the ceiling of a home, collected as part of the Human Speechome Project [Roy et al., 2006].

Open-ended language understanding is a challenging problem, requiring the ability to sense complex events in video and map those events to natural language descriptions. To make progress on this problem I focus on spatial language search of trajectories which are automatically extracted from video recorded by stationary overhead cameras. The system takes as input a natural language query, a database of surveillance video from a particular environment, and the locations of non-moving objects in the environment. It parses the query into spatial description clauses. SDCs automatically extracted for a sample query are shown in Figure 6-5. Using a model for the joint distribution of a natural language query (represented as a sequence of SDCs) and trajectories, it finds $p(query, track)$ for each trajectory in the corpus and returns video clips sorted by this score, performing ranked retrieval. The system can find video clips that match arbitrary spatial language queries, such as "People walking down the hall into the living room," by leveraging the decomposition of the language into SDCs and background knowledge found in large online databases. Video clips



"The person walked from the couches in the living room to the dining room table."

"The woman entered the dining room from the living room."

"She walks from the hallway into the dining room and stands by the side of the dining room table that is nearest to the kitchen."

"The person walked from the couch in the living[*sic*] to the dining table in the dining room."

"The person enters the dining room from the living room and goes to the table near the entrance to the kitchen."

"She starts in the living room and walks to in front of the desk."

"The person enters the dining room from the stairway or living room area. She goes to the long side of the table nearest to the kitchen doorway."

"The person walks from the left-bottom side of the dining room table over tot he[*sic*] shelves."

Figure 6-1: Frames from a video clip in the corpus, together with descriptions for that clip. The person's starting location is marked in green, the ending location in red, and the path they took in blue. The same trajectory converted to global coordinate system is overlayed on the floor plan of the house in the right.

for a sample query returned by the system are shown in Figure 6-3.

In order to perform video retrieval, the system needs to compute $p(query, clip)$ for each video clip in the database. The system takes as input a database of trajectories automatically extracted from surveillance video recorded in a particular environment, and the locations and geometries of observed objects in the environment, such as couches, chairs and bedrooms. Although automatic object detection could be used in conjunction with overhead cameras, this was not the focus of this paper. Thus, it was practical to manually label the locations of a small set of non-moving objects in each camera. These explicitly labeled landmarks are used to bootstrap resolution of landmark phrases that appear in queries; the system infers the locations of unobserved objects based on observed ones.

6.1 Open-Ended Corpus

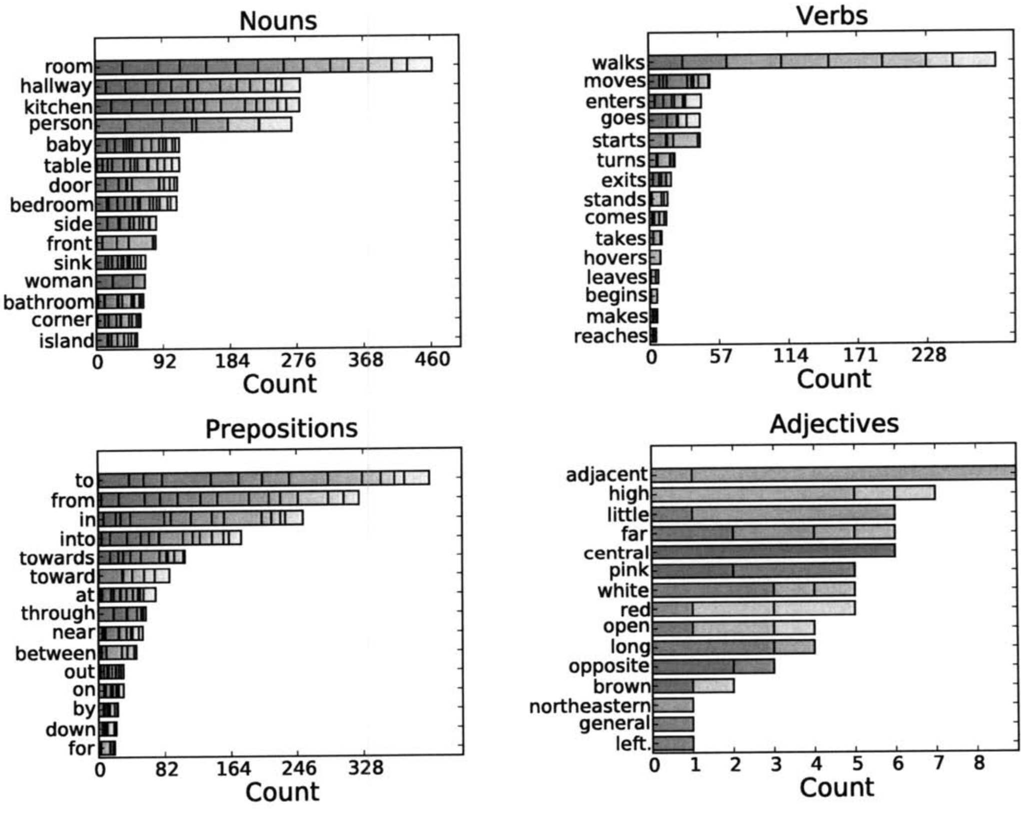


Figure 6-2: Histogram showing the most frequent words in the open-ended video retrieval corpus for various parts of speech.

In order to train and evaluate models for the meanings of spatial relations, I collected a corpus of natural language descriptions of video clips. This corpus gives insight into the types of descriptions humans use when labeling video. The corpus consists of a person's movement in a video clip paired with a natural language description of the movement.

The video corpus used was collected as part of the Human Speechome Project [Roy et al., 2006]. Video was collected from eleven ceiling-mounted cameras which were installed in a home as part of an effort to understand a child's language acquisition. Video was recorded for more than three years, capturing a large longitudinal record of a family's activity in their home. Because cameras and microphones were installed unobtrusively in the ceiling, the recording became part of their routine rather than a more invasive shorter-term data collection.

To collect natural language descriptions of activity paired with video, annotators watched short clips with the location of a person marked in each frame of the clip. They were asked to write down a natural language description of the motion of a person in the clip. Annotators were asked to skip the clip if there was a tracking failure, or if they could not write a description for the clip.

A person's location was marked automatically by a tracker. The tracker used in this part of the work was developed by George Shaw. Movement traces, or tracks are generated using a motion-based tracking algorithm. Pixels representing movement in each video frame are collected into dense patches, or particles, with these particles providing probabilistic evidence for the existence of a person. These particle detections allow models to be built up over time. By correlating each such model from frame to frame, the system can efficiently and robustly track the movement of people in a scene.

In the first, more restricted variant of this task, described in Chapter 3, annotators saw the sentence "The person is going" and were asked to complete that sentence with whatever made the most sense to them. Here in a more open-ended version, annotators were asked to describe the person's motion so that a different annotator could draw the person's trajectory on a floor plan of the house, using only the description. These descriptions were much more free-form and complex than the phrase-based descriptions collected as part of Chapter 3.

I used a corpus of tracks that was generated to understand lexical acquisition of the child in the Human Speechome Project. Word births, or the first time a particular word was used by the child, are located in time by semi-automatic annotation of the audio portion of the HSP dataset [Roy et al., 2006]. The tracker was launched 10 minutes in advance of each word birth and was allowed to run until 10 minutes following the word birth, creating a 20 minute window into the context surrounding the first use of the word by the child. This method was followed for 50 word births, producing a corpus of approximately 1000 minutes of tracks throughout the entire house.

In order to collect a corpus of natural language descriptions of tracks, the database of tracks was sampled to extract two datasets of fifty tracks. The tracks were created by sampling 10 random 2.5 second clips, 10 random 5 second clips, 10 random 10 second clips, 10 random 20 second clips, and 10 random 40 second clips from the first five word births. Clips were constrained to end at least two meters from where they started, to ensure that the clip contained at least some motion. Otherwise, many tracks consisted of a person sitting at a table or on a couch, and never moving. The first dataset allowed clips to overlap in time in order to collect more than one description from the same person for the same track at different granularities. The second dataset had no overlaps in time to collect a more diverse database. The clips were collected randomly from all the cameras in the main floor of the house, but each individual clip was from a single camera.

Fourteen annotators were recruited from the university community to view each clip and describe the activity of a person in the clip. Annotators viewed each clip, with the location of the person being tracked marked by a large green dot on each frame of the clip. They were instructed to describe the motion of the person in the video so that another annotator could draw their trajectory on a floor plan of the house. During the initial instructions, I showed each annotator a floor plan of the house to familiarize them with the layout and how the scenes from each camera connected. I did not ask them to restrict their language in any way, but rather use

Query	Avg. Precision
People coming out of the bathroom.	0.833
People walking into the baby's bedroom.	0.917
People walking down the hall.	0.967
People walking around the table in the living room.	1.000
People walking into the kitchen.	1.000
People walking out of the kitchen.	0.704
People walking from the hallway door, around the island, to the kitchen sink.	0.583
Mean Average Precision	0.858

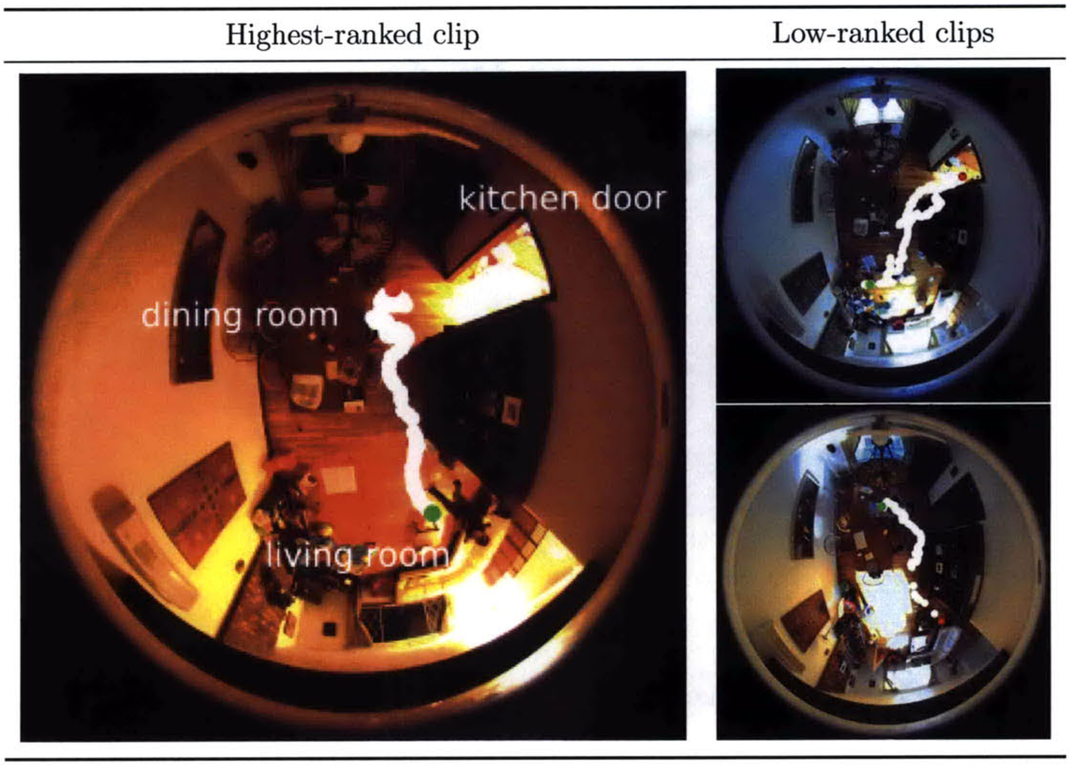
Table 6.1: Average precision for representative queries.

whatever language they felt appropriate to describe the person's motion. At times the automatic person tracker made errors. Annotators were instructed to mark tracks where the automatic person tracker made significant errors. They skipped on average 6.5/50.0 tracks, implying the tracker worked fairly well most of the time.

Some sample descriptions from the corpus are shown in Figure 6-1. Annotators' vocabulary was constrained only by the task. They used full sentences, whatever landmark objects they felt were appropriate, and were not instructed to use a particular set of verbs or spatial relations. A histogram of the fifteen most frequent words for different parts of speech appears in Figure 6-2. Annotators used mostly nouns and spatial relations to specify landmarks, and relatively few adjectives.

6.2 Results

I report the model's performance in different configurations to analyze the importance of different spatial relations to the system's overall performance. In order to



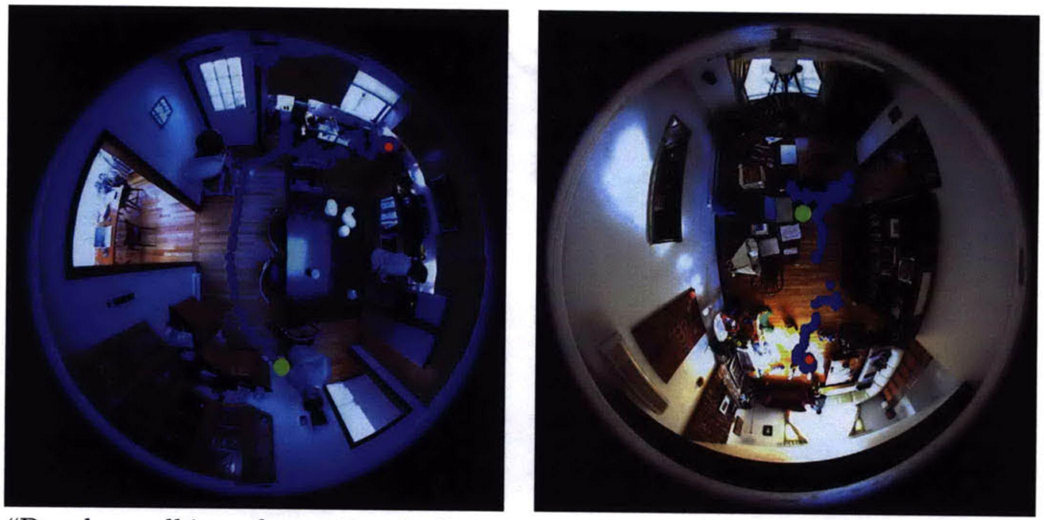
Highest-ranked clip

Low-ranked clips

Figure 6-3: Results from the system for the query "from the couches in the living room to the dining room table." The person's start location is marked with a green dot; the end location is marked with a red dot, and their trajectory is marked in white.

assess the system's performance in different configurations, I developed an evaluation metric based on a ranked retrieval task. For each natural language description in the corpus, the system created a dataset of 10 tracks, containing the original track the annotator saw when creating the description and nine other random tracks. The system computed $p(query, track)$, using the description as the query for all ten tracks and sorts the clips by this score. I report the average rank of the original clip in this list over all 696 descriptions in the corpus. If the description is treated as a query, the original clip should have a high rank in this list, since it should match the query better than the other random clips. A system that ranks randomly out of ten would have an average rank of 5.5, marked with a dotted line on the graph. I report 90% confidence intervals in all graphs.

For the first experiment, I compared the system's performance with and without



"People walking from the hallway door, around the island, to the kitchen sink."

"People walking into the living room, coming from the dining room."

Figure 6-4: Frames from highly ranked clips for two queries.

spatial relations. Without spatial relations, it uses only the landmark field to match the video clip to the person's trajectory. The results in Figure 6-8 indicate that spatial relations significantly improved the performance of the overall system. With spatial relations, 406 (58.3%) of descriptions were ranked one or two; without spatial relations, only 39.9% were ranked one or two. This result shows that spatial relations capture an important part of the semantics of the trajectory descriptions.

Next, Figure 6-7 shows the performance of the system when run on only those descriptions in the corpus that contain the labeled spatial relations. Only spatial relations for which I have a model, and for which more than 10 examples appeared in the corpus are shown. In almost all cases, the model of the spatial relation decreases the average rank, improving retrieval performance. Although this result is often not significant there is a consistent positive effect for spatial relations; that the overall trend is significant can be seen in Figure 6-8.

This automatic evaluation metric does not work perfectly. For example, the classifier for "down" performs well, as measured on its cross-validated training set and in the results presented in Table 6.1. However, in the corpus, "down" was almost

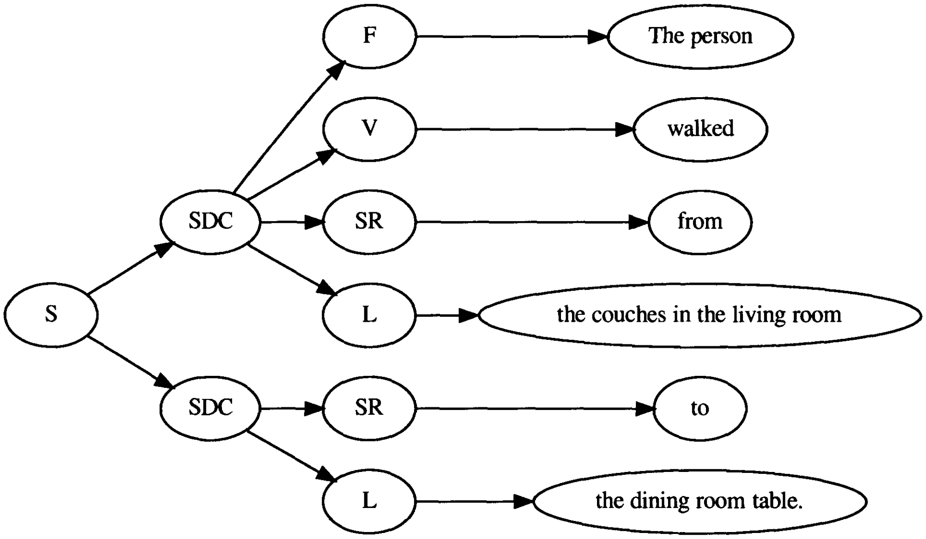
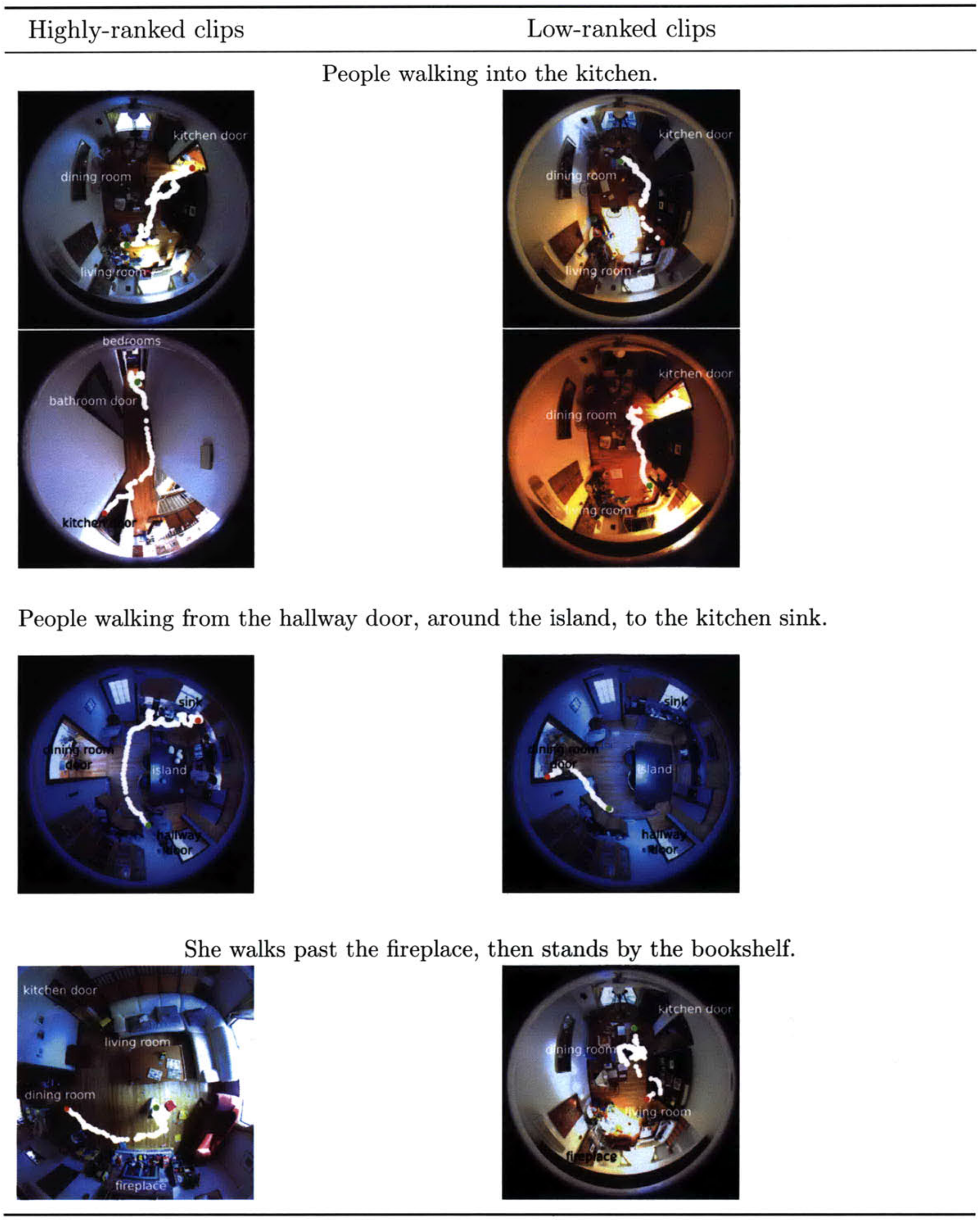


Figure 6-5: SDCs automatically extracted for the sentence "The person walked from the couches in the living room to the dining room table," created by an annotator for the clip shown in Figure 6-3. An SDC consists of a figure F, a verb V, a spatial relation SR, and a landmark L.



Highly-ranked clips

Low-ranked clips

People walking into the kitchen.

People walking from the hallway door, around the island, to the kitchen sink.

She walks past the fireplace, then stands by the bookshelf.

Figure 6-6: Clips returned for various queries. High-scoring examples are shown on the left; low-scoring examples are shown on the right.

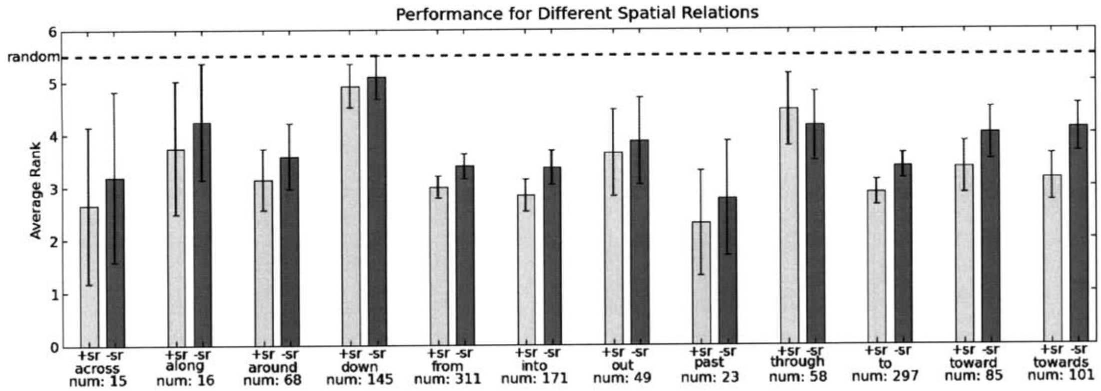


Figure 6-7: Results with and without models for each spatial relation, using only descriptions in the dataset that contained the particular relation. Error bars are 90% confidence intervals.

always used in the context of "down the hallway." Furthermore, people rarely do anything in the hallway except walk down it. When they do, the model for "down" ranks these examples as more likely. Moreover, the overall performance of "down" is poor according to this metric because there were many examples of people walking down the hallway in the corpus of video clips. These clips were ranked higher than the original clip because they also matched the natural language description.

Next I investigated the contribution of individual features in the models for the meanings of spatial prepositions to the system's overall performance. I did this comparison on the subset of descriptions that contained that particular spatial relation. The results for "towards" are shown in Figure 6-9. Here it can be seen that the model using all features outperforms any model trained with a single individual features, showing that information is being fused from multiple features to form the semantics of the spatial relation.

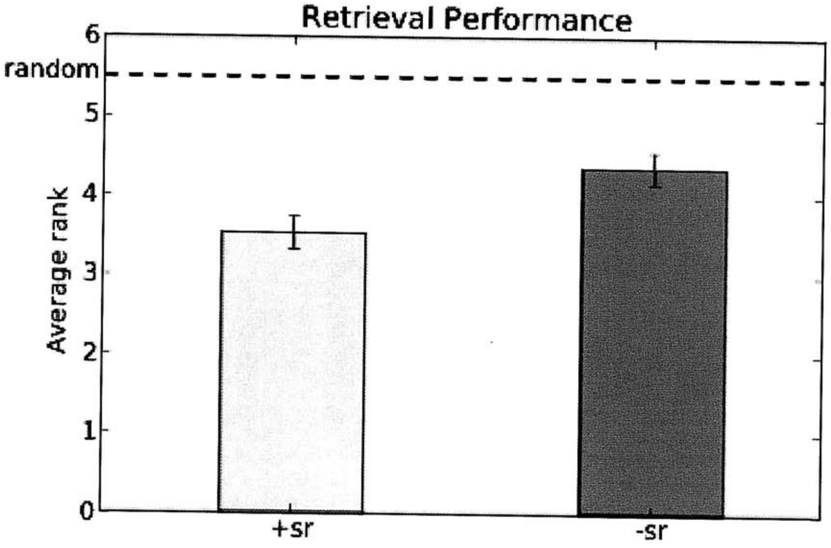


Figure 6-8: Results with and without models for the semantics of spatial relations, on the entire corpus.

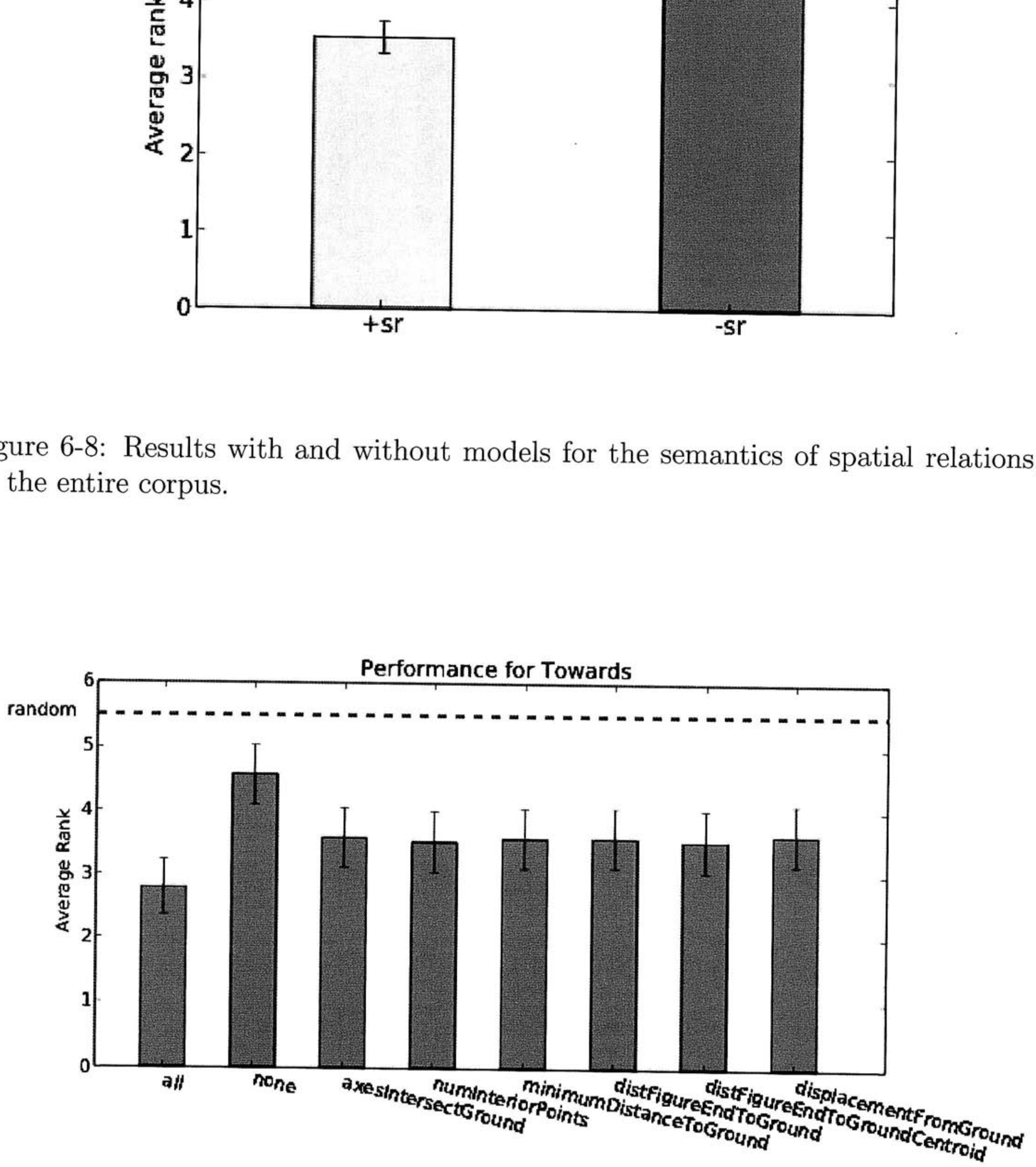


Figure 6-9: Results with for "towards," showing all features, no features, and each feature individually. Error bars are 90% confidence intervals.

6.2.1 Performance on Particular Queries

In order to assess the performance of the end-to-end system, I ran it on several representative queries on a dataset of fifty tracks. Tracks were returned in order according to $p(query, track)$. I report performance using average precision, a measure commonly used to report the performance of ranked retrieval systems [Manning and Schütze, 1999]. It is computed by averaging precision at rank r for all $r \in R$, where R is all relevant documents for a particular query. This metric captures both precision and recall in a single number and reflects how well the system is ranking results. The highest possible value in the corpus is 1, if all relevant documents are ranked before irrelevant documents; the lowest value is 0.02, if there was only one relevant document that was returned last. In order to compute this metric, I made relevance judgments for each query: for each of the fifty tracks in the dataset I annotated whether it matched the query or not. Results are reported in Table 6.1.

I chose queries that seemed to reflect particular information needs. For example, doctors monitoring the health of elders are interested in renal failure, and might be concerned with how frequently they use the restroom or how frequently they enter the kitchen to eat. Social scientists or interior designers might be curious about how people use the space and how to lay it out better.

Average precision is generally quite high. These results indicate that the system is successfully retrieving video clips from a large dataset of trajectories for representative queries that are useful for answering real-world questions.

Finally, Figure 6-6 shows example of high and low-scoring trajectories returned by the system for various queries. This concretely shows that the system is correctly making fine distinctions in the semantics of various queries.

Chapter 7

Space and Time

The system described so far has an important limitation: it cannot handle time-varying commands. This chapter describes joint with with Thomas Kollar that addresses this limitation: extensions that can handle verbs such as "follow" and "meet." For a command such as "Follow me to the kitchen," it generates a sequence of actions that corresponds to the desired motion through the environment. This problem is challenging because the verbs have a complex internal event structure, and the robot must respond to the person's movements when obeying the command.

To address this problem, we need a two-pronged approach: first a new inference mechanism is required to replan in the event of changes in the environment. Second, new models are required to capture the meanings of words such as verbs that depend on changes in the environment.

Planning is done using a cost function defined in terms of the log likelihood of the model described in Chapter 4. Once a cost function is defined, the robot can be controlled via closed-loop planning. At each timestep, the robot computes the lowest-cost sequence of actions and executes the corresponding action, enabling it to dynamically respond to changes in the environment for commands such as "Follow the person."

The second component required is models for verbs such as "meet" and "follow." Chapter 3 described supervised classifiers to model the meanings of spatial relations in the natural language command and ground landmark noun phrases using back-

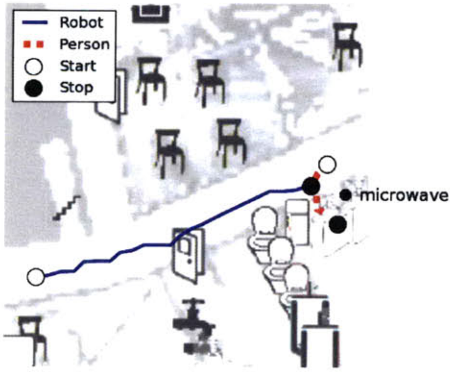


Figure 7-1: Top scoring plan for "Meet the person at the kitchen." The system searches over action sequences the robot could take, and possible trajectories for the person. It outputs the most likely pair of trajectories for the robot and person.

ground knowledge about object co-occurrence information mined from a large corpus of tagged images. Here I describe time-based verb models that can handle both figures and landmarks that move.

7.1 Inference

Given a natural language command, the algorithm performs forward search to find the sequence of actions that minimizes the cost function. At each timestep, the algorithm computes a set of candidate plans and landmarks that maximizes the joint distribution of actions, states, and observations. Dynamically re-planning at each timestep enables the robot to react to a changing environment.

Evaluating plans with the model is expensive and make several approximations to the model to make this tractable. For spatial relations sr, landmarks l, and states s_i, $p(sr, l|s_1 \dots s_T)$ is approximated as $\arg\max_{pathlets} p(sr, l|pathlet)$, where pathlets are the set of path subsets on a particular candidate path. In this way the spatial relation and landmark computations can be cached for each pathlet in the map (only a few hundred), greatly speeding up inference. This approximation was not made for verb models, because the paths of the robot and person were much more important for the ranking of verbs. In addition, for verbs the landmark is sometimes an object in the environment, and other times a person. I used a keyword spotting algorithm to decide if a landmark phrase represented a person, and otherwise backed off to the inference based on Flickr co-occurrence statistics. For verbs which take a landmark, rather than marginalizing over all possible landmarks g, the system picks the location which maximizes $p(v_t, l_t, g_t|s_1 \ldots s_T)$ as $\arg\max g_t p(v_t, l_t|s_1 \ldots s_T, g_t)p(g_t)$. The prior on $p(g_t)$ encourages the landmark to be closer to the robot.

7.1.1 Verbs

Verbs are one of the most complex and important parts of natural language directives. They define events that take place between one or more entities in the world, imposing a rich internal structure. Syntactically, they form the core of the sentence, relating the rest of the phrases together to form a coherent event. This section describes a methodology for learning models for the meanings of verbs of motion. When training the verbs, the system must face both moving (e.g. people) and non-moving (e.g. objects) landmarks. Unlike static geometric features, verbs such as "follow the person" require models that can dynamically react to changes in the person's trajectory.

For a transitive verb of motion such as "follow," the system decomposes the motion trajectories of the person following and the person being followed, together with limited information about the environment: $p(v_k|f_k, l_k, s_1 \ldots s_T)$. The figure ($f_k$) and landmark ($l_k$) are given to the model as a sequence of locations tagged with times. This distribution may need to be conditioned on the entire state sequence since, for example, "bring" may need to have access to the part of the path where person and robot met as well as the trajectory afterward in order to rank an example trajectory appropriately.

To learn this distribution I created Boolean features that have a truth value at each instant in the event are created:

MovingTowards(landmark, figure) IsMoving(landmark)
IsVisible(figure, landmark) IsClose(figure, landmark)
MovingTowards(figure, landmark) IsMoving(figure)

I also used conjunctions of all features. The system samples feature values at each time slice of the event and computes the probability of a feature value being true

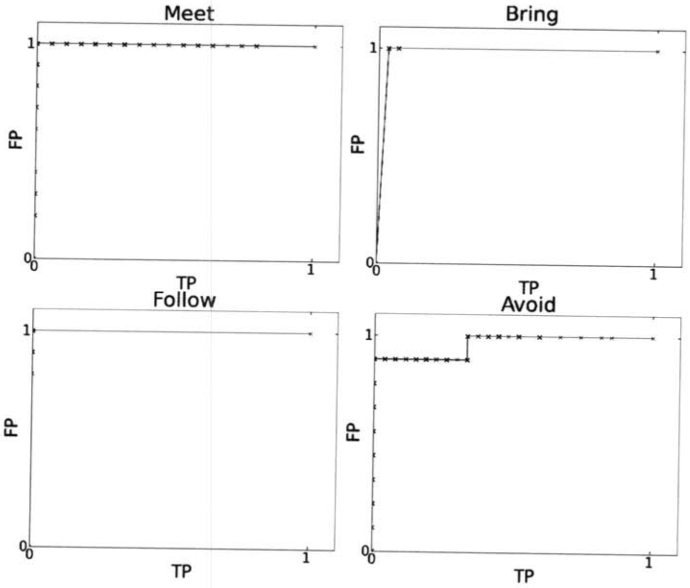


Figure 7-2: ROC curves showing the performance of distributions for various verbs when they are treated as classifiers. TP is true positives, and FP is false positives.

during the event. In addition, I used features corresponding to the path as a whole, such as the distance between the figure and landmark at the beginning and end of the event. All of these features were used as input to a naive Bayes distribution which was trained using a corpus of labeled examples, following the same methodology as the one used for spatial relations. Verbs can be composed with any of the spatial relations and landmarks, according to Equation 4.7, which allows for the composition of novel commands that the robot has never seen before. In addition, since the features used to learn the verbs are scale-invariant and generalize to new paths, this component will generalize to new environments.

Standard machine learning techniques suffice to learn good verb models. As the domain is expanded to more complex verbs, more powerful inference mechanisms are required to capture more of the time-based structure of the verb. Performance curves for the verbs trained on a corpus of labeled examples is shown in Figure 7-2.

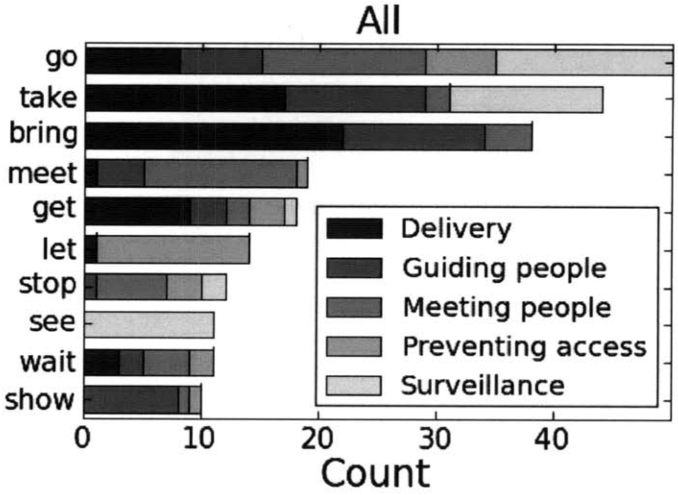


Figure 7-3: Most frequent verbs from an open-ended set of natural language commands given to robots.

Go	Follow	Avoid	Meet	Bring	**Overall**
90%	80%	78%	70%	29%	**69%**

Table 7.1: Accuracy of our algorithm for various verbs.

7.2 Evaluation

The current system takes as input a natural language command and an initial position for the person and robot. At each timestep the system infers a plan and takes an action. An example output of one step of the inference appears in Figure 7-1, for "Meet the person at the kitchen." The system searches over possible paths of the robot and the person, finds the most likely one according to the model, and takes the corresponding action. It does this at each timestep, so the overall output is a trajectory for the robot through the environment, re-planning at each step in response to the person's actions.

I present a quantitative evaluation of our system's performance using a corpus of natural language commands compared with a person's actions when following those commands. The system used the inference algorithm to control the robot's activity, given the same information a person had when creating the corpus. Table 7.1 reports the fraction of commands that a researcher judged that the robot got correct. Most of the commands used a map populated by hand with the labels of certain classes of objects.

The verb "bring" performs much worse than the others in the test set. This disparity is for two reasons. First, "bring" involves more complex event structure than other verbs: the robot must first approach the person, then take them somewhere else. This structure is difficult for a simple feature-based classifier to model. Second, because "bring" events tend to be longer, the model might not have been able to search far enough ahead in time to find a successful plan. Optimizing and parallelizing the system to enable it to search deeper in the plan space could alleviate this problem.

This work points the way towards a mechanism for handling time-based commands. It works for verbs like "follow" and "avoid" but breaks down on more structured commands. However the framework described in this thesis for decomposing natural language into sub-components, and defining a distribution that maps between language in the world provides a jumping-off point for building these richer, time-based models.

Chapter 8

Conclusion

This thesis describes a system for understanding natural language commands and descriptions about events in the world. Much work remains to be done to extend the thesis to handle more realistic types of dialog and embed the work in real-world applications.

8.1 Future Work

Spatial language understanding is a challenging problem with many real-world applications. This thesis addressees aspects of that problem and provides a base to build future work. Key areas of extension include moving from paths to plans for describing more complex, hierarchical activity, developing learning mechanisms that require less supervision, and adding dialog capabilities to the system.

The biggest challenge that lies ahead is extending beyond spatial language to more general vocabulary and more faithful meaning representations. This expansion includes adding richer models of the meanings of words, that incorporates functional geometry and nuances such as "just past," as well as expanding the scope of language, from higher level actions and plans to more abstract spatial metaphor.

8.1.1 Goal-based Language

Higher-level goal-based queries and commands such as "Get me a soda" or "Show me people eating a snack" directly map to the goals of a person interacting with a language understanding system; a system that understands these sorts of commands would be more useful than one that merely processes spatial directives. To make this concrete, I asked an annotator to describe a person's activity in video by completing the sentence "The person is..." rather than "The person is going..." as in the restricted video retrieval corpus. Figure 8-2 shows a histogram of the resulting annotations. Spatial descriptions appear, but also many high-level descriptions such as "preparing food," "tidying up" and "cooking."

Related, to characterize the types of things a human might say to a robot, an open-ended corpus of natural language commands was collected. Subjects were asked to imagine a mobile robot that can recognize people and objects but cannot manipulate objects or engage in dialog. Subjects wrote down natural language commands they might give to a robot for five different scenarios: delivery, guiding people, meeting people, preventing access, and surveillance. Seven subjects were robotics researchers, and five were annotators hired from the MIT community. Example commands from this corpus include:

- "Wait by the staircase next to 391, and bring Susan to my room when she comes in."

- "Please go to the question mark and wait for Oussama. When he gets here, bring him back here."

Figure 8-3 shows a histogram of the most frequent verbs in the corpus, broken down by command type. Commands in this corpus are extremely challenging, because of the complex syntactic structure, wide variety of vocabulary, and detailed knowledge of the environment.

To handle these types of language, it is necessary to develop a system that can map between language and more general actions, rather than just movement. There will be a correspondingly larger space to search when finding the best mapping between language and actions and events. Finding the right search space is critical; fast, approximate inference techniques will make the search more tractable. The methodology described here, of breaking down language into its components, modeling each component separately, and building it back up, can provide guidelines for structuring this search.

Chapter 7 described a planning-based inference mechanism for finding a plan that matches a command such as "Follow the person to the kitchen." This framework points the way towards more general commands to robots such as "Pick up the box and load it into the truck" or "Take a picture of the person on the couch." Handling these types of commands requires defining a space of plans to search over that is rich enough to capture the semantics of language, but small enough to be tractable. One of the largest implementation issues that came up over and over in this work was finding ways to reduce the size of the search space in order to perform tractable inference. Applications include CSAIL's robotic forklift, developed as part of the Agile Robotics In Process project [Correa et al., 2010] and shown in Figure 8-1. The goal of this project is to develop a robot capable of loading and unloading cargo in a warehouse environment. Other manipulator robots such as Willow Garage's PR2 and the MDS have arms and hands capable of manipulating objects, and even folding towels [Maitin-Shepard et al., 2010].

As robots move out of the lab, it is important to create natural, easy-to-use interfaces to support human-robot interaction. Language is a compelling modality for human-robot interaction because in the best case it requires no training or adaptation on the part of the user. Making a robot that can engage in naturalistic dialog with a human is a challenging problem, requiring both the ability to understand natural language utterances as well as the ability to participate in natural language dialog. The system described in this thesis is really about understanding a part of a complete dialog interaction; embedding it in a complete dialog interface is required for language interfaces to move out of the lab.

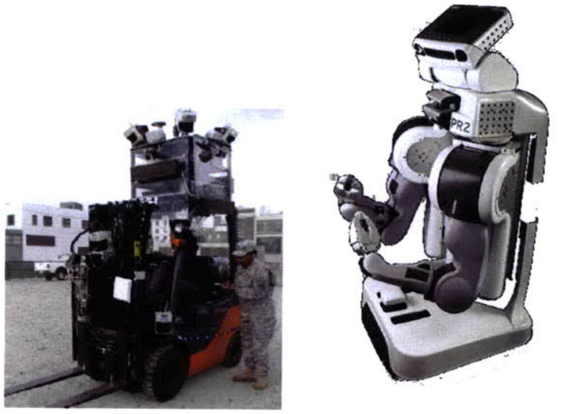


Figure 8-1: CSAIL's robotic forklift, and Willow Garage's PR2. These robots have the capability to manipulate objects. Language interfaces require richer vocabularies and a higher-level set of actions.

8.1.2 Functional Geometry

It appears that many spatial relations do not rely on just the geometrical information of the figure and ground, but also require functional information. Coventry and Garrod [2004] have shown that judgements of situations such as "in the bowl" depend not just on geometric relations but also functional relations: whether the object is supported by the bowl or by the rope, as shown in Figure 8-4. Extending the system to handle these features would enable a more faithful model of the semantics of spatial prepositions.

8.1.3 Spatial Metaphor

Spatial language appears in many non-spatial contexts. For example, the sentence "Through the years, you've never let me down" contains spatial analogies to both time ("through the years") and emotion ("let me down"). Boroditsky [2000] has shown that space and time share conceptual structure, and that the domain of time is shaped by structures from spatial reasoning. This result suggests that it should be possible to leverage spatial language understanding in order to understand these types of metaphor. If a spatial model of an abstract domain was created, then the system described in this paper could be used to reason about it, but there are many

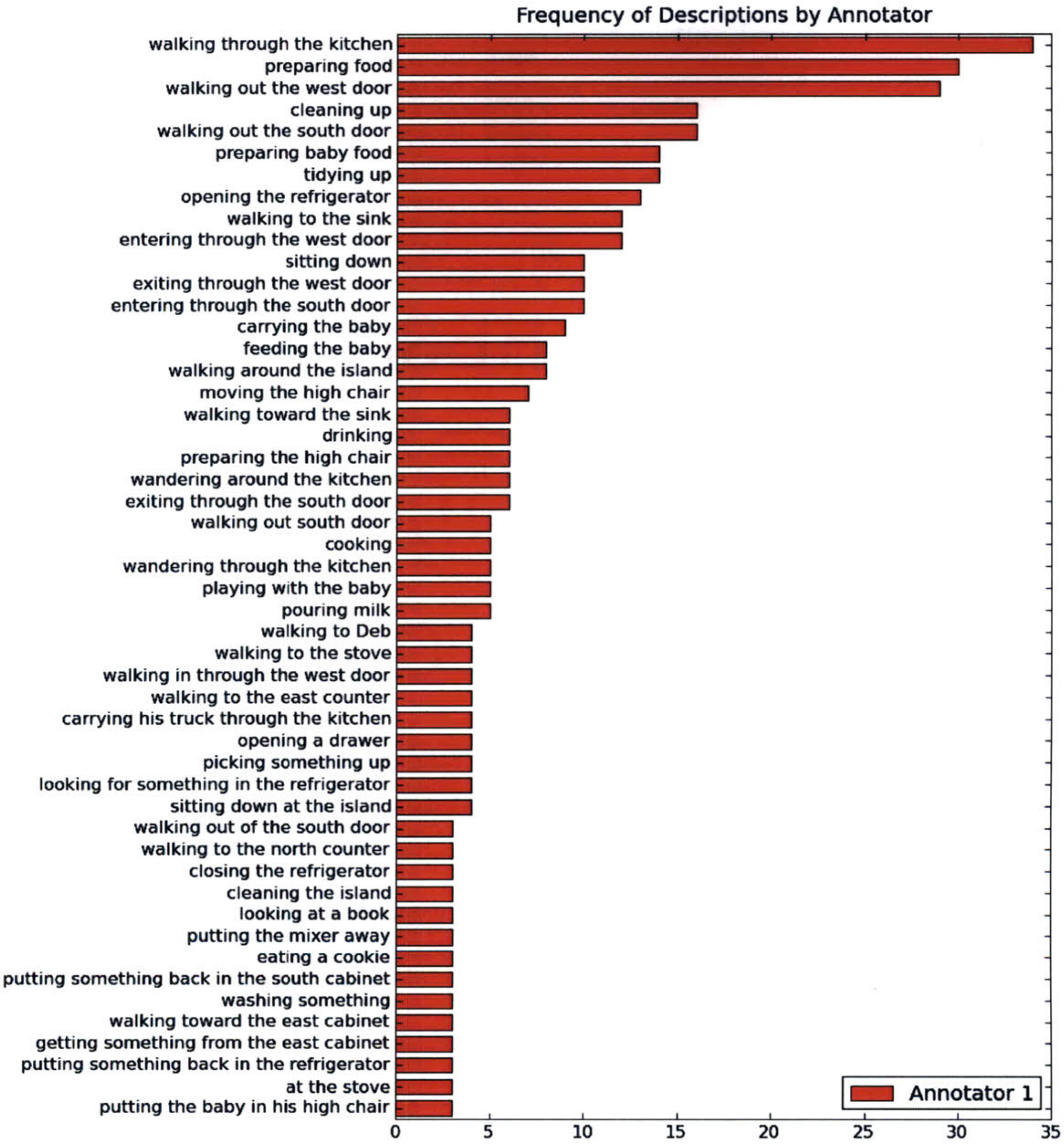


Figure 8-2: Descriptions created by an annotator watching video clips and completing the sentence "The person is ..."

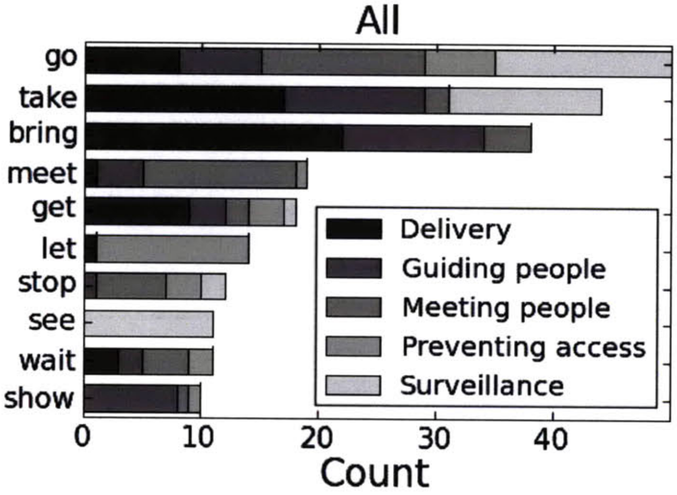


Figure 8-3: Most frequent verbs from an open-ended set of natural language commands given to robots.

nuances. This could enable a system that can understand non-spatial language by analogy to spatial language.

8.1.4 Map Interfaces

With the new, nearly universal availability of location information, and the impending revolution in robotics, there are many opportunities for this work to have both near-term and long-term impact. Devices with access to GPS-based location information are now everywhere. An improved, dialog-based voice interface to a GPS navigation system would enable users to safely use the technology while they are driving, since it requires neither their hands nor their eyes.

8.1.5 GPS Logs

GPS is now available on almost every cell phone and in many cars. As more and more GPS data is recorded, there is an increasing need for interfaces for searching the data. Applications areas include information surveillance and reconnaissance for the military, and law enforcement.

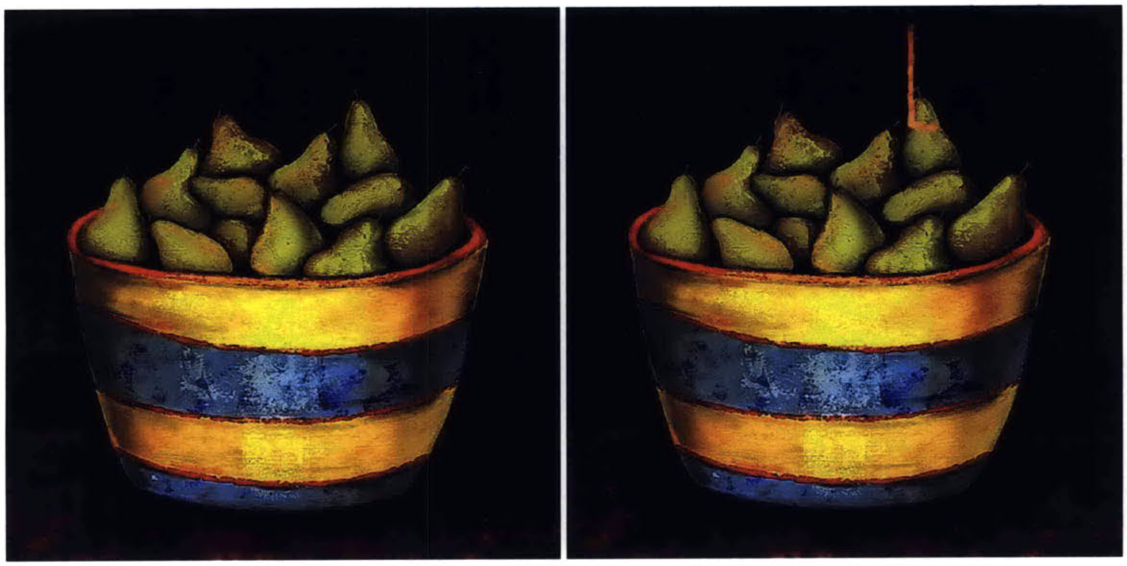


Figure 8-4: Judgements of whether "The pear is in the bowl" depend on whether the bowl is controlling the movement of the pear.

8.2 Contributions

The work described in this thesis describes how to make systems that understand spatial language in real-world contexts. Expanding beyond spatial language understanding to richer language, and embedding it in dialog are the next steps towards building robust language understanding systems that work in the real world.

The key scientific contribution of this thesis is a model of spatial semantics that enables a system to understand and use spatial language in real-world domains. Spatial prepositions in English are defined in terms of a set of features extracted from the two-dimensional geometry of a scene. I applied this lexicon of spatial relations to two real-world problems: natural language video retrieval and natural language direction understanding. Overall the system performs quite well. For direction understanding, the system correctly follows between 50% and 60% of directions in the corpus, compared to 85% for human performance. Furthermore, I show that spatial relations improve performance when doing local inference to follow the directions. When doing video retrieval with the open-ended corpus, the system effectively retrieves clips that match natural language descriptions: 58.3% were ranked in the top two of ten in a retrieval task, compared to 39.9% without spatial relations. This effort shows the effectiveness of the features and provides an opportunity to analyze their performance in order to study which ones perform best. The thesis advances the state of the art in natural language understanding and grounding by connecting spatial language to real-world domains.

Bibliography

J. F. Allen. Maintaining knowledge about temporal intervals. *Commun. ACM*, 26 (11):832–843, 1983.

D. Bailey. *When Push Comes to Shove: A Computational Model of the Role of Motor Control in the Acquisition of Action Verbs.* PhD thesis, 1997.

J. Black, T. Ellis, and D. Makris. A hierarchical database for visual surveillance applications. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1571– 1574, 2004.

L. Boroditsky. Metaphoric structuring: Understanding time through spatial metaphors. 2000.

G. Bugmann, E. Klein, S. Lauria, and T. Kyriacou. Corpus-Based robotics: A route instruction example. *Proceedings of Intelligent Autonomous Systems*, pages 96— 103, 2004.

A. Correa, M. R. Walter, L. Fletcher, J. Glass, S. Teller, and R. Davis. Multimodal interaction with an autonomous forklift. In *Proceeding of the 5th ACM/IEEE International Conference on Human-robot Interaction*, pages 243–250, Osaka, Japan, 2010. ACM.

K. Coventry and S. Garrod. *Saying, Seeing, and Acting.* Psychology Press, Routledge, UK, 2004.

J. Demsar and B. Zupan. Orange: From experimental machine learning to interactive data mining. Technical report, Faculty of Computer and Information Science, University of Ljubljana, 2004. URL `http://www.ailab.si/orange`.

J. Dzifcak, M. Scheutz, C. Baral, and P. Schermerhorn. What to do and how to do it: Translating natural language directives into temporal and dynamic logic representation for goal management and action execution. In *IEEE International Conference on Robotics and Automation (ICRA-2009)*, pages 4163–4168, 2009.

M. Fleischman, P. DeCamp, and D. Roy. Mining temporal patterns of movement for video content classification. In *Proceedings of the 8th ACM SIGMM International Workshop on Multimedia Information Retrieval*, 2006.

G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with Rao-Blackwellized particle filters. *IEEE Transactions on Robotics*, 23(1):34–46, 2007.

K. Hsiao, S. Tellex, S. Vosoughi, R. Kubat, and D. Roy. Object schemas for grounding language in a responsive robot. *Connection Science*, 20(4):253–276, 2008.

Y. A. Ivanov and C. R. Wren. Toward spatial queries for spatial surveillance tasks. In *Pervasive: Workshop Pervasive Technology Applied Real-World Experiences with RFID and Sensor Networks (PTA)*, 2006.

R. S. Jackendoff. *Semantics and Cognition*, pages 161–187. MIT Press, 1983.

B. Katz, J. Lin, C. Stauffer, and E. Grimson. Answering questions about moving objects in surveillance videos. In M. Maybury, editor, *New Directions in Question Answering*, pages 113–124. Springer, 2004.

J. D. Kelleher and F. J. Costello. Applying computational models of spatial prepositions to visually situated dialog. *Computational Linguistics*, 35(2):271–306, June 2009.

K. Kinzer. Tweenbots: Robot/people art. http://www.tweenbots.com, 2009.

A. Klippel, H. Tappe, L. Kulik, and P. U. Lee. Wayfinding choremes–A language for modeling conceptual route knowledge. *Journal of Visual Languages & Computing*, 16(4):311–329, 2005.

T. Kollar and N. Roy. Utilizing object-object and object-scene context when planning to find things. In *IEEE International Conference on Robotics and Automation*, 2009.

T. Kollar, S. Tellex, D. Roy, and N. Roy. Toward understanding natural language directions. In *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*, 2010.

T. Kudo. CRF++: Yet another CRF toolkit. http://crfpp.sourceforge.net, 2009.

B. Landau and R. Jackendoff. "What" and "where" in spatial language and spatial cognition. *Behavioral and Brain Sciences*, 16:217–265, 1993.

M. Levit and D. Roy. Interpretation of spatial language in a map navigation task. *Systems, Man, and Cybernetics, Part B, IEEE Transactions on*, 37(3):667–679, 2007.

X. Li, D. Wang, J. Li, and B. Zhang. Video search in concept subspace: A text-like paradigm. In *Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 603–610, Amsterdam, The Netherlands, 2007. ACM.

T. Lochmatter, P. Roduit, C. Cianci, N. Correll, J. Jacot, and A. Martinoli. Swistrack - A flexible open source tracking software for multi-agent systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008.

M. Macmahon. Walk the talk: Connecting language, knowledge, and action in route instructions. *In Proc. of the Nat. Conf. on Artificial Intelligence (AAAI)*, pages 1475—1482, 2006.

J. Maitin-Shepard, J. Lei, M. Cusumano-Towner, and P. Abbeel. Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding. In *International Conference on Robotics and Automation (ICRA)*, Anchorage, Alaska, USA, May 2010.

C. D. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing.* MIT Press, 1999.

C. Matuszek, D. Fox, and K. Koscher. Following directions using statistical machine translation. In *HRI 2010: Proc. of the 5th Int'l Conf. on Human-Robot Interaction*, Osaka, Japan, 2010. ACM Press, ACM Press.

M. Naphade, J. Smith, J. Tesic, S. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and J. Curtis. Large-scale concept ontology for multimedia. *Multimedia, IEEE*, 13(3): 86–91, 2006.

T. Regier and L. A. Carlson. Grounding spatial language in perception: An empirical and computational investigation. *Journal of Experimental Psychology. General*, 130(2):273–98, June 2001. PMID: 11409104.

T. Regier and M. Zheng. Attention to endpoints: A cross-linguistic constraint on spatial meaning. *Cognitive Science*, 31(4):705, 2007.

T. P. Regier. *The Acquisition of Lexical Semantics for Spatial Terms: A Connectionist Model of Perceptual Categorization.* PhD thesis, University of California at Berkeley, 1992. Ph.D. thesis.

W. Ren, S. Singh, M. Singh, and Y. Zhu. State-of-the-art on spatio-temporal information-based video retrieval. *Pattern Recognition*, 42(2):267–282, Feb. 2009.

D. Roy, R. Patel, P. DeCamp, R. Kubat, M. Fleischman, B. Roy, N. Mavridis, S. Tellex, A. Salata, J. Guinness, M. Levit, and P. Gorniak. The Human Speechome Project. In *Proceedings of the 28th Annual Cognitive Science Conference*, pages 192–196, 2006.

J. M. Siskind. Grounding the lexical semantics of verbs in visual perception using force dynamics and event logic. *J. Artif. Int. Res.*, 15(1):31–90, 2001.

M. Skubic, D. Perzanowski, S. Blisard, A. Schultz, W. Adams, M. Bugajska, and D. Brock. Spatial language for human-robot dialogs. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 34(2):154–167, 2004. ISSN 1094-6977.

L. Talmy. The fundamental system of spatial schemas in language. In B. Hamp, editor, *From Perception to Meaning: Image Schemas in Cognitive Linguistics*. Mouton de Gruyter, 2005.

S. Tellex and D. Roy. Towards surveillance video search by natural language query. In *Conference on Image and Video Retrieval (CIVIR-2009)*, 2009.

H. S. Thompson, A. Anderson, E. G. Bard, G. Doherty-Sneddon, A. Newlands, and C. Sotillo. The HCRC map task corpus: natural dialogue for speech recognition. In *Proceedings of the workshop on Human Language Technology*, pages 25–30, Princeton, New Jersey, 1993. Association for Computational Linguistics. ISBN 1-55860-324-7. URL `http://portal.acm.org/citation.cfm?id=1075677`.

A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *Information Theory, IEEE Transactions on*, 13(2):260–269, 1967. ISSN 0018-9448.

J. Vlahos. Welcome to the Panopticon. *Popular Mechanics*, 185(1):64, 2008. ISSN 00324558.

Y. Wei, E. Brunskill, T. Kollar, and N. Roy. Where to go: Interpreting natural directions using global inference. In *IEEE International Conference on Robotics and Automation*, 2009.

T. Winograd. *Procedures as a representation for data in a computer program for understanding natural language.* Thesis, Massachusetts Institute of Technology, 1970. Ph.D. thesis.

A. Yoshitaka, Y. Hosoda, M. Yoshimitsu, M. Hirakawa, and T. Ichikawa. Violone: Video retrieval by motion example. *Journal of Visual Languages and Computing*, 7:423–443, 1996.