

A Taxonomy of Situated Language in Natural Contexts

by

George Macaulay Shaw

B.F.A., Boston University (2008)

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

Master of Science in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2011

© Massachusetts Institute of Technology 2011. All rights reserved.

Author _____
Program in Media Arts and Sciences
August 5, 2011

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

Accepted by _____
Mitchell Resnick
LEGO Papert Professor of Learning Research
Academic Head
Program in Media Arts and Sciences

A Taxonomy of Situated Language in Natural Contexts

by

George Macaulay Shaw

The following person served as reader for this thesis:

Thesis Reader _____
Deb Roy
Associate Professor of Media Arts and Sciences
MIT Media Lab

A Taxonomy of Situated Language in Natural Contexts

by

George Macaulay Shaw

The following person served as reader for this thesis:

Thesis Reader_____

Pawan Sinha
Associate Professor of Vision and Computational Neuroscience
Massachusetts Institute of Technology

A Taxonomy of Situated Language in Natural Contexts

by

George Macaulay Shaw

The following person served as reader for this thesis:

Thesis Reader _____

Yuri Ivanov
Principal Software Engineer
Heartland Robotics

A Taxonomy of Situated Language in Natural Contexts

by

George Macaulay Shaw

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
on August 5, 2011, in partial fulfillment of the
requirements for the degree of
Master of Science in Media Arts and Sciences

Abstract

This thesis develops a multi-modal dataset consisting of transcribed speech along with the locations in which that speech took place. Speech with location attached is called situated language, and is represented here as spatial distributions, or two-dimensional histograms over locations in a home. These histograms are organized in the form of a taxonomy, where one can explore, compare, and contrast various slices along several axes of interest.

This dataset is derived from raw data collected as part of the Human Speechome Project, and consists of semi-automatically transcribed spoken language and time-aligned overhead video collected over 15 months in a typical home environment. As part of this thesis, the vocabulary of the child before the age of two is derived from transcription, as well as the age at which the child first produced each of the 658 words in his vocabulary.

Locations are derived using an efficient tracking algorithm, developed as part of this thesis, called 2C. This system maintains high accuracy when compared to similar systems, while dramatically reducing processing time, an essential feature when processing a corpus of this size. Spatial distributions are produced for many different cuts through the data, including temporal segments (i.e. morning, day, and night), speaker identities (i.e. mother, father, child), and linguistic content (i.e. per-word, aggregate by word type).

Several visualization types and statistics are developed, which prove useful for organizing and exploring the dataset. It will then be shown that spatial distributions contain a wealth of information, and that this information can be exploited in various ways to derive meaningful insights and numerical results from the data.

Thesis Supervisor: Deb Roy

Title: Associate Professor of Media Arts and Sciences, Program in Media Arts and Sciences

Acknowledgments

It has been an honor and a privilege to work in such incredible company as I've enjoyed at the Media Lab. MIT has been an absolutely amazing environment in which to revel in the joy of science and I must thank the entire MIT community, past and present for the difference MIT has made to my life.

Thanks to my advisor Deb for his guidance and endless wisdom; and for believing in me, making my time at the Media Lab possible. Also thank you to my thesis readers, Yuri and Pawan, without whom I would have been lost before I even got started. Thanks to my entire research group Cognitive Machines, and particularly to Rony, Philip, and Brandon who pushed me to try harder, do more, and think more creatively than I ever thought possible. Finally, thanks Karina for keeping me human and (mostly) sane.

Most of all, I thank my wife Victoria and two sons Laszlo and Renatto for inspiring, encouraging, assisting, and tolerating me during this long, exciting journey. Their patience and support has been indispensable.

Contents

Abstract	7
1 Introduction	17
1.1 Goals	20
1.1.1 Multi-modal Understanding	20
1.1.2 Situated Language: establishing context for everyday language use	20
1.1.3 Practical Applications	23
1.1.4 Ancillary goals	24
1.2 Methodology	25
1.2.1 How to situate language: a system blueprint	25
1.2.2 Taxonomy: Exploring a Large Dataset	26
1.2.3 Visualization	27
1.3 Contributions	28
2 Dataset	31
2.1 The Human Speechome Project	32
2.2 2C: Object Tracking and Visual Data	34
2.2.1 Overview	34
2.2.2 The Tracking Problem	34
2.2.3 Input Component	38
2.2.4 Low-level Feature Extraction	39
2.2.5 Object tracking	42
2.2.6 Performance	48
2.2.7 Tuning	51
2.3 Transcription and Speaker ID	54
2.4 Processed Data	55
2.4.1 Processed Tracks	55
2.4.2 Child’s Vocabulary and Word Births	57
2.4.3 Situated Utterances	59
2.4.4 Spatial Distributions	60
2.5 Summary	61
3 Taxonomy	63
3.1 Overview	63
3.2 Schema	63

3.3	Visualizations	66
3.3.1	Heat Maps	67
3.3.2	Difference Maps	67
3.4	Statistics	67
4	Exploration and Analysis	77
4.1	Activity Types	77
4.1.1	Speech vs. Movement	77
4.2	Speech content	78
4.2.1	Target Words vs. All Speech	79
4.2.2	Spatial groundedness	80
4.2.3	Clustering	83
4.3	Identity	86
4.4	Temporal slices	87
4.5	Age of Acquisition correlation	88
5	Conclusions	97
5.1	Contributions of This Work	97
5.2	Future Directions	98
A	Data for Target Words	101
B	Visualizations for Target Words	127
	References	161

List of Figures

1-1	Basic system design	18
1-2	System with human operator	19
2-1	Overview of Dataset	32
2-2	Views from 4 of the 11 cameras	33
2-3	Reconstructed 3D view of the home [6]	33
2-4	Motion-based tracking. Detections are clustered into objects that share coherent velocities.	43
2-5	Tracking pipeline: raw video, motion detection and aggregation, tracking .	45
2-6	Sample Movement Traces	55
2-7	Track processing pipeline	56
2-8	Old vs. New Word Births	58
2-9	Word birth verification plot	59
2-10	Summary of Dataset Processing	61
3-1	Overview of Dataset and Taxonomy	64
3-2	Taxonomy schema	66
3-3	Sampled and observed KL-divergence	72
3-4	The relationship between count and observed KL-divergence	73
3-5	Difference Browser	76
4-1	Heat maps: (L to R) all activity, all activity plotted on log scale, all activity with 1000mm bins	78
4-2	Heat maps: (Clockwise from top left) all speech, all speech plotted on log scale, all speech with 1000mm bins, social hotspots	79
4-3	Difference map showing speech vs. all activity	80
4-4	Target word heat maps	81
4-5	Top 150 Words by Ripley's K	82
4-6	Snapshots taken during utterances containing "ball" in the location associated with "ball"	83
4-7	K-means clustering by KL-divergence and Ripley's K	84
4-8	Difference maps for (clockwise from top left): child, father, nanny, mother .	88
4-9	Difference maps for (clockwise from top left): morning, daytime, evening, night	89
4-10	Predictor accuracy as a function of sample count threshold	90
4-11	KL-residual correlation with AoA	91
4-12	Prediction error vs. actual age of acquisition	94

List of Tables

2.1	Accuracy and precision comparison	50
2.2	Runtime stats for tracking components	51
2.3	Runtime stats for tracking module steps	51
A.1	Data for words in the child’s vocabulary	102

Chapter 1

Introduction

Data goes in, answers come out.

It is by now obvious that large datasets will be a hallmark of the coming years. Decreasing costs of storage and processing as well as improved techniques for analysis are sparking the generation of more and more datasets that until recently would have been unthinkable large. Of particular interest are those datasets that bring disparate data types together: social media linked to television, retail transaction data linked to surveillance video, or time-aligned speech and video are just a few examples. These multi-modal datasets allow the researcher to explore not only each modality in isolation, but more importantly to explore and understand the linkages and alignments between modalities.

These datasets are only useful if we can ask questions of them and expect to receive an accurate, relevant answer. We'd like to put in some data, possibly a lot of data, and get back an answer that allows us to make a business decision, pursue science, or achieve some other goal.

If a picture is worth a thousand words, a video must be worth a million.

An important modality to consider, particularly as collection and storage costs are driven

lower, is video. There are vast amounts of information contained in video streams; information that is often difficult to process and analyze, but that is extremely dense and useful when processed successfully. Video is also a very natural datatype for people to work with. Watching video corresponds easily to normal visual perception, and analysis outcomes are often more intuitive and readily understood because the modality itself is so familiar. While this natural understanding of video and related data is clearly an advantage for a researcher working on the project, there are ancillary benefits in that outside researchers or other stakeholders in a project are able to assimilate and utilize the data easily as well.

When video is aligned with other data and viewed in aggregate form, analysis can bring about insights that would otherwise have been opaque even to a dedicated researcher spending countless hours manually watching footage - the nature of the patterns in aligned multi-modal data and the varying scales at which these patterns occur often make insights subtle and difficult to ascertain without robust computational methods.

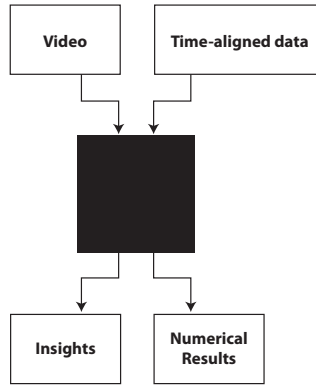


Figure 1-1: Basic system design

Aligning video with other data sources is central to the work in this thesis. Many of the methods described here were developed using several different datasets with dissimilar “other” data in addition to video. Consider one such dataset consisting of video from a typical surveillance system in a retail environment in addition to transaction data from that retail location. At the algorithmic level, building, managing, exploring, and deriving insights from such a dataset is nearly identical to performing those tasks on a corpus of video taken in a home,

aligned with transcription of the speech in that home, as is the focus here. These similarities provide generality for the approaches described here - it is my goal that this work be relevant across many domains and disciplines.

This thesis describes one implementation of the more general system (the “black box” in Figure 1-1) that accepts two time-aligned data sources and produces insights and numerical results.

Generating the types of insights and results that are useful in any particular domain automatically is a hard problem. Computers are not yet capable of true undirected exploration and analysis, so instead I bring a human operator into the design of the system as a collaborator. This notion of human-machine collaboration was first put forth by J.C.R. Licklider in [13], where he describes a “very close coupling between the human and the electronic members of the partnership” where humans and computers cooperate in “making decisions and controlling complex situations without inflexible dependence on predetermined programs.” Over 50 years after Lickliders famous paper, this approach rings true now more than ever and serves to frame the work described here.

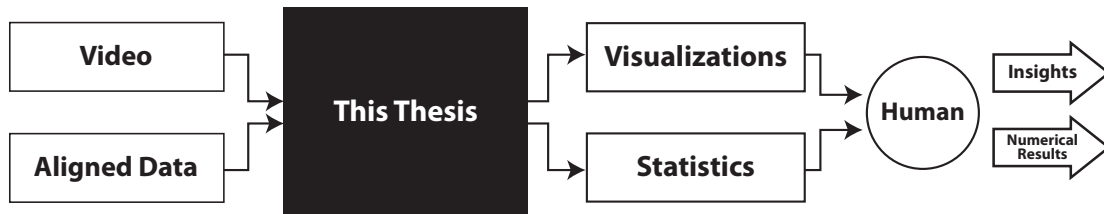


Figure 1-2: System with human operator

In the system described here, the human operator accepts a wealth of data from the computer in the form of visualizations and statistics, parses this data, and derives results appropriate to the task at hand, possibly providing feedback to the system in order to revise and iterate. Figure 1-2 shows the revised system design, now with an operator in place. This thesis focuses on the black box, or the part of the system that processes multiple data sources and generates user-friendly output data.

1.1 Goals

1.1.1 Multi-modal Understanding

Integrating information across modalities is the key to true understanding. It has been argued that multimodal sensing is at the core of general intelligence [17], with some even going so far as to say consciousness itself is the result of “integrating information” across modalities [34]. From an information-theoretic perspective, adding information from additional modalities can only increase understanding (intuitively, notice that you can always just ignore the new information if it provides no help and by ignoring unhelpful information your overall understanding has remained the same).

I seek to explore multi-modal analysis from two perspectives: from the standpoint of building a system that uses information across modalities in order to derive accurate, meaningful insights; and from the standpoint of a child learning to speak, who uses linguistic information in addition to contextual information in order to begin to understand language.

These are clearly different, but complementary problems. Carver Mead famously said that *“If we really understand a system we will be able to build it. Conversely, we can be sure that we do not fully understand a system until we have synthesized and demonstrated a working model.”* By building a system that attempts to integrate what is seen with what is said, it is reasonable to hope that we can gain some insights into how a child begins to integrate what he is seeing with the language he is hearing.

1.1.2 Situated Language: establishing context for everyday language use

Labeling the things in our world is at the core of human intelligence. Our success as a species is due in large part to our ability to use language effectively, and to connect that language to the physical world - in other words, to label discrete objects and concepts. In order to understand the cognitive processes at the heart of our language use, we must

understand the context in which language takes place in addition to understanding the linguistic features. This work attempts to shed light on a few of the patterns associated with language use in a natural environment and some of the properties of those patterns.

There has been significant work in the grounding of language in perception [26], an idea that provides linguistic scaffolding to enable infants and intelligent machines to begin to connect symbolic representations of language to the real world. This connection of symbols to real world perception is crucial to understanding how language use comes about, and provides a foundation on which we can build richer and more complex notions of communication.

Situated language is language for which a context has been established.

Everyday language exists in a rich context that provides the listener with countless clues as to the underlying meaning of a linguistic act. This context must be taken into account when attempting to understand language at any more than a surface level, and includes all of the various properties of the environment in which the language occurs. Nowhere is this context more important than in everyday speech, where much meaning is unspoken and implied, to be gleaned via context by the listener. Contextual cues would often provide useful clues for understanding the language used in the home. Knowing that there is a bag of flour nearby, for example, provides essential clues as to the meaning of the phrase “please hand me the flour,” which would be interpreted differently if there were a bouquet of roses on the table.

To understand context, we might consider modeling all of the myriad cues present during a speech act. These cues would include the entire array of visual stimuli, identities of participants (speakers and listeners), and temporal features (time of day, day of week, etc.), as well as details about the activity taking place at the time. To fully model context, we would also need to include complete histories of all participants (for example, relevant context for a conversation could include a previous conversation with the same participants), current psychological states, audible cues, and environmental features such as temperature

and wind. Clearly, such a model is computationally infeasible, therefore we must focus on relevant bits of this context, and on computationally tractable proxies for these bits of context.

One such useful proxy for environmental context is the location of the speech act. The location of a speech act contains a wealth of information about the context surrounding that speech act in the form of an abstraction of such information. By knowing that an utterance has taken place in the kitchen, for example, we are implicitly examining information about the visual context of that utterance. The kitchen contains visual cues x , y , and z , therefore all speech taking place in the kitchen can be tied on some level to x , y , and z because x , y , and z are part of the context in which language in the kitchen is immersed.

Temporal features also provide important context that can stand in for many other complex cues in our non-linguistic environment. The various activities that we participate in provide crucial pieces of information about what is said during these activities. These activities often occur at regular times, so by examining language through the lens of its temporal context, we obtain a useful proxy for the types of activities that occur at that time. When taken together with spatial context, temporal context becomes even more powerful. The kitchen in the morning, for example, stands in for the activity “having breakfast,” a context that is hugely helpful to the understanding of the language taking place in the kitchen in the morning.

Participant identity is the final contextual cue utilized here, and the one that stands in for the most unseen information. The identity of a participant can encompass the entire personal history of that participant: consider an utterance for which we know that the speaker is person X . If we have aggregated speech from person X in the past, then we can determine that this person tends to conduct themselves in certain ways - displaying particular speech and movement patterns and so on. We don’t need to know *why* person X does these things, it is enough that we can establish a proxy for person X ’s history based on their past actions, and that we can now use this history in current analysis of person

X’s speech.

In this work, context is distilled to a compact representation consisting of the location in which a speech act occurred, the identity of the participants, and the time at which the utterance was spoken. In this thesis, I intend to show that even this compact form of context provides valuable information for understanding language from several perspectives: from that of an engineer hoping to build systems that use language in more human-like ways, and from that of a cognitive psychologist hoping to understand language use in human beings.

1.1.3 Practical Applications

Understanding language deeply has long been a goal of researchers in both artificial intelligence and cognitive psychology. There has been extensive research in modeling language from a purely symbolic point of view, and in understanding language use by statistical methods. This work is limited, however, as words are understood in terms of other words, leading to the kind of circular definitions that are common in dictionaries. There has been interest, however, in grounding language use in real world perception and action [26], a direction that hopes to model language in a manner that more closely resembles how people use language. This work essentially says that “context matters” when attempting to understand the meaning of a word or utterance, and more specifically that visual perception is an important element of context to consider.

Understanding and modeling the non-linguistic context around language could provide huge practical benefits for artificial intelligence. Especially as datasets grow larger and corpora such as the Human Speechome Project’s become more common, access to the data necessary for robust non-linguistic context estimation will become simple for any well engineered AI system. However, a clearer understanding of how this context should be integrated must be developed.

As a concrete example, consider automatic speech transcription. Modern systems rely

on both properties of the audio stream provided to them and immediate linguistic context in order to perform accurate, grammatically plausible transcription. If we were to give a system a sense of the non-linguistic context around a language act, we might expect transcription accuracy to improve dramatically. Consider a human performing language understanding - listening to a conversation, in other words. If this person were to attempt to perform transcription based solely on the audio it receives from its sensors, we would expect accuracy to be low. Adding some knowledge of grammar would help considerably, but accuracy would still be below the levels that we would expect from a real person performing this task. But by allowing the person to leverage non-linguistic context (as would be the case when the person understands the language being transcribed and so can bring to bear all of their experience in order to disambiguate the meaning of the language and therefore the content of the language itself) we would expect accuracy to be near perfect. It is clear, then, that providing this context to an AI system would allow for far more accurate transcription as well.

From the point of view of human cognitive psychology, analysis of the context surrounding language development will lead to better understanding of the role of this context, which in turn will lead to deeper understanding of the mechanisms by which children come to acquire language. There are many potential applications of such insights, one example being the facilitation of language learning in both normally developing and developmentally challenged children.

1.1.4 Ancillary goals

There are several aspects of this work that relate to other goals: areas that are not primary foci of the work, but that I hope to make some small contribution to. As this work is centered around an extensive dataset, the broader goal of increased understanding of engineering and effective analysis of large datasets is important. These datasets present problems that simply do not exist in smaller datasets - problems that have been overcome in Human Speechome Project analysis.

This work also holds visualization as a central element, and so hopes to add to the discourse around effective visualization, particularly scalable visualization techniques that can be applied to large, complex datasets.

Finally, machine vision is a key component of the construction of the situated language dataset described in this thesis. The problems faced in performing vision tasks on this data are central to most cases where vision is to be applied to a large dataset, and the solutions presented here are both unique and applicable to a wide range of vision problems.

1.2 Methodology

1.2.1 How to situate language: a system blueprint

Consider a skeletal system that is capable of situating language. This system must possess, at a minimum, a means of representing language in a way that is manipulable by the system itself. While there are many forms of language that can be represented and manipulated by a computer system, here I focus on basic symbolic language - English in particular.

It is possible to imagine many schemes for determining the locations of people. Such schemes might rely on any of a variety of sensors, or any number of methods for deriving person locations in even a simple video-style sensor (such as what we have here). We might attempt to find people in video by matching shape templates, or by looking at pixel motion patterns, or by performing tracking of all objects over time and determining later which tracks represent interlocutors in a speech act. Any of these methods share the common output of deriving conversational participants' locations at the time of the conversation.

From the representation of language, this system provides the statistical backing around which to begin linguistic understanding. But from the locations of participants, this system derives context for the language. And then, assuming such a system is capable of repre-

senting time and that it records temporal information for the language it represents, the system can also provide temporal context.

Our basic system requirements are therefore:

1. A symbolic representation for language

Represented here as written English

2. A means of deriving and representing participant locations

Represented here as coordinates in Euclidean space relative to a single home, derived by performing person tracking in time-aligned recorded video

3. A way to represent and record temporal information for speech acts

Represented here as microsecond timestamps aligned across video and audio data (and therefore locational and transcript data)

1.2.2 Taxonomy: Exploring a Large Dataset

The best known taxonomies are those that classify nature, specifically the Linnean Taxonomy, which classifies organisms according to kingdoms, classes, orders, families, and so on. Carl Linnaeus set forth this taxonomical representation of the world in his 1735 work *Systema Naturae*, and elements of this taxonomy, particularly much of the classification of the animal kingdom, are still in use by scientists today.

It has been argued that Darwin's theory of evolution owes a great deal to his detailed taxonomical explorations of animals [40]. Darwin is thought to have spent many years building his taxonomy, noting features, similarities, and differences between various animals. This objective, unbiased classification of organisms without specific research goals may have been crucial to Darwin's understanding of the evolutionary mechanisms he later set out in *Origin of Species*.

This thesis sets out to create a taxonomy of natural language use over the course of 15 months in the home of one family. The taxonomy consists of information about the location of the things that were said in the home, segmented across 3 dimensions of interest, with many data points and organizational metrics related to these segmentations. I attempt to categorize and structure various properties of situated language in ways that are likely to provide meaning in understanding that language. Furthermore, I attempt to frame this exploration through the lens of acquiring language, as language acquisition can be thought of as the most basic form of (and a useful proxy for) language understanding. The creation of this taxonomy, like Darwin’s creation, has led and will continue to lead to new insights and research directions about how language is used in day to day life.

1.2.3 Visualization

The dataset presented here is significantly complex - it represents much of the home life of a normal family over the course of 15 months, and as such contains much of the complexity and ambiguity of daily life. There is no quick and easy way to gain understanding of this dataset - exploration and iteration is essential to slowly building up both intuition and numerical insight into the data. Visualization is a good way to explore a dataset of this size. Visualization benefits greatly from structure, however, and the taxonomy detailed here provides that structure.

Visualization of quantitative data has roots that stretch back to the very beginnings of mathematics and science [35]. Visualizing mathematical concepts has been shown to be essential to learning and understanding [8], a result that points to the fundamental notion that quantitative information is represented visually in ways that are more easily assimilated and manipulated by people [9, 22].

Abstraction has been an undeniably powerful concept in the growth of many areas of science, especially computing. Without abstraction, programmers would still be mired in the intricacies of machine code and the powerful software we take for granted would have been

impossible to create. Human beings have finite resources that can be brought to bear on a problem. By creating simpler, higher level representations for more complex lower level concepts, abstraction is an essential tool for conserving these resources. Visualization can be thought of as a kind of abstraction, hiding complexity from the viewer while distilling important information into a form that the viewer can make sense of and use.

This work heavily leverages the power of visualization as a foundation of its analysis. Several fundamental visualization types are central to the work, with other ad-hoc visualizations having been undertaken during the course of research and development of the systems described.

By treating numerical and visual data as qualitatively equal lenses into the same complex data, we can think of the output of our system as truly multi-modal. Furthermore, such a system leverages the strengths of both modalities - numerical data and mathematical analysis provides precision and algorithmic power, while visualization provides views into the data that a person can reason about creatively and fluidly, even when the underlying data is too complex to be fully understood in its raw form.

1.3 Contributions

Primary contributions of this work are to:

- Demonstrate the construction of a large dataset that spans multiple modalities
- Develop novel visualization methods, with general applicability to any “video + aligned data” dataset
- Utilize visualization and statistical approaches to construct a taxonomy of the patterns present in the normal daily life of a typical family
- Understand behavioral patterns segmented along various dimensions including time

of day, identity, and speech act, and show how these patterns can be explained and analyzed in a data-driven way

- Using statistical properties of the patterns derived above, show that non-linguistic context is correlated with the age at which the child learns particular words and provide a possible explanation for such correlation.

Chapter 2

Dataset

The dataset described here is comprised of situated language, or language for which temporal information as well as the locations and identities of participants are known. From this situated language data, we can generate spatial distributions representing aggregate language use along various dimensions of interest (i.e. temporal slice or the use of some particular word).

We begin with raw, as-recorded video and audio. Audio is then semi-automatically transcribed and video is processed by machine vision algorithms that track the locations of people. Tracks are smoothed and merged across cameras, and transcripts are tokenized along word boundaries and filtered to remove non-linguistic utterances and transcription errors. Tracks and transcripts are then joined by alignment of the timestamps in each. Transcripts with corresponding location information (points) are called situated language. Finally, situated language data is distilled into spatial histograms. See Figure 2-1.

2.1 The Human Speechome Project

The Human Speechome Project [28], undertaken with the goal of understanding child language acquisition, sought to record as much of a single child’s early life as possible, capturing a detailed record of the child learning to speak in a natural setting.

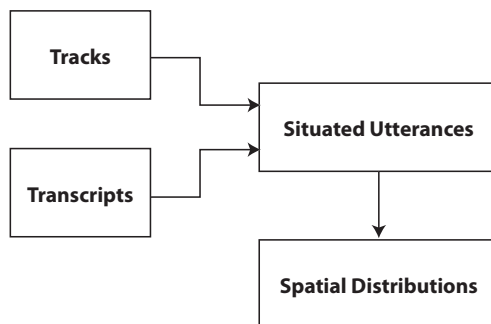


Figure 2-1: Overview of Dataset

Video was collected from eleven cameras installed in ceilings throughout a typical home. Views from four cameras are shown in Figure 2-2. All occupants of the home were recorded, including the mother, father, nanny, and child. Recording took place only while the child was awake and at home, and occupants were able to suspend recording at any time. Cameras were identical and were placed in order to provide maximum coverage of the home’s living spaces. Each high dynamic range camera was equipped with a fisheye lens and provided on-board jpeg compression. Cameras were connected via Ethernet to a central control system that ensured synchronicity across cameras as well as accurate frame-level timestamps. Audio was recorded using 14 boundary-layer microphones, each connected to the same control system as the video cameras. Microphones were positioned in order to provide maximum coverage of the audible environment in the home. Care was taken to ensure that the audio was suitably timestamped and was synchronized with the video streams. Further details about the recording and storage of data can be found here [4].

Recording resulted in approximately 90,000 hours of multi-channel video and 140,000 hours

of audio recorded over the course of 3 years. We estimate that the total recorded data represents approximately 75% of the child's waking life. Here I focus on the 15 month period during which the child was 9-24 months of age.

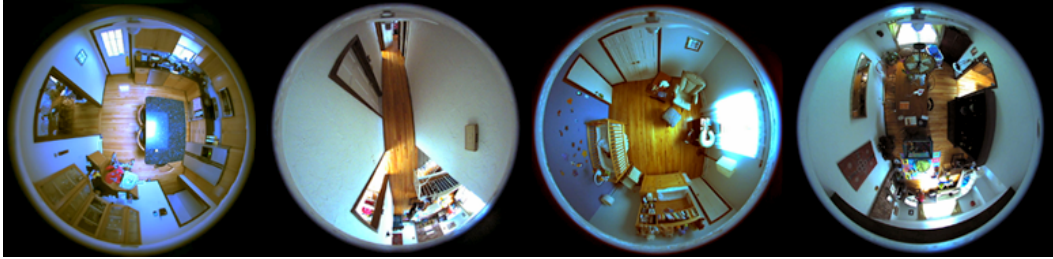


Figure 2-2: Views from 4 of the 11 cameras

Figure 2-3 shows a reconstructed 3D view of the home, visualized using the Housefly [6] system. In this view, we can see the various rooms clearly. Clockwise from top left, we have the dining room, kitchen, bathroom (no recording), master bedroom (no recording), guest bedroom, child's bedroom, and living room.



Figure 2-3: Reconstructed 3D view of the home [6]

2.2 2C: Object Tracking and Visual Data

2.2.1 Overview

“There is a significant body of literature surrounding the interpretation of human behavior in video. A common thread in all of this work is that tracking is the very first stage of processing.” [12]

Object tracking is an integral part of this work, and the tracking mechanisms described here are a key contribution of this thesis. In particular, this software tracks objects with accuracy and precision comparable to the state of the art, while performing these tasks an order of magnitude faster than other equally powerful systems.

The 2C vision system is a flexible framework for performing various vision tasks in a variety of environments. 2C provides a powerful foundational API, enabling a developer to extend the capabilities of the system easily via custom modules that can be chained together in arbitrary configurations. 2C contains a set of interfaces for input, processing, and output modules, data structures and protocols for communication between those modules, and infrastructure necessary for robust operation. Here I focus on one application of this system: tracking people in the HSP dataset. Therefore, from here forward 2C will refer not to the system as a whole, but to the particular configuration focused on efficient person tracking.

2.2.2 The Tracking Problem

At its simplest, a tracking system must implement some attention allocation scheme (“*what to track?*”) and some method of individuating targets (“*where is the thing I saw in the last frame?*”).

More formally:

We have a set of features f_t derived from video input at time t and a (possibly null) set of

existing objects O_{t-1}

From these we need to generate the set of objects O_t .

For algorithm $a(\cdot)$ the tracking problem can be described simply as:

$$O_t = a(f_t, O_{t-1})$$

Of course, this leaves out a lot of detail. What is $a(\cdot)$? How do we describe O ? What are the features f ? Do we aggregate t across many frames, solving globally, or derive each O_t individually?

Tracking problems range from very easy (imagine tracking a moving black object on a white field - even the simplest algorithm solves this problem well) to very difficult (consider tracking individual bees in a hive [36] - the most sophisticated approaches will still make errors). There are also cases where tracking requires higher level inference - to decide whether to track a baby in his mother's arms, for example, requires knowledge beyond what vision can provide and so even the most sophisticated algorithms will fail in these cases.

There are several key differentiators in this particular tracking task that define the direction of much of 2C's design. The following considerations were most important in the design of 2C:

- **The nature of the input video.** HSP video contains huge lighting variation at many temporal resolutions (i.e. day vs. night or lamps being turned on and off). A robust, unsupervised approach is needed that can work in a variety of lighting conditions.
- **The size of the corpus.** Even moderately sophisticated approaches to object tracking can require extensive computational resources that would make processing the 90,000 hours of video in the HSP corpus infeasible.

- **The analysis needs of the project.** The expected use of the output of the system dictates how many design decisions are evaluated. In the case of current HSP analysis, it was more important to provide accurate moment-to-moment views of occupancy than long contiguous tracks, a consideration that resulted in several important design decisions.

Based on the considerations listed above, it was determined that a highly adaptive system was needed that would perform object tracking in as efficient a manner as possible, while still maintaining accuracy at the point level.

Many tracking approaches appear in the literature [41], and many of these have been implemented within the 2C architecture. Of particular interest here are efficient approaches that might be combined as building blocks in the design of a larger system such as 2C.

When considering the design of an efficient object tracking system, it is natural to look to an existing system that performs this task well: the human visual cortex. In the human visual cortex, we have a system that performs near perfect tracking in almost all situations, but whose operation we have only a cursory understanding of. Work such as [23, 30, 32] has attempted to characterize the fundamental mechanisms for object tracking by studying humans' ability to track generic objects. A variety of insights and constraints have come out of this experimental work. Of particular importance in the context of this system are the results that explore the types of features people use and don't use when tracking objects. People can track robustly even if shape and color features of an object change over time [30] - this result points to coherent velocity as the primary means by which object tracking is done.

Intuition, however, would seem to indicate that shape and color do play a role in tracking at least part of the time. It doesn't seem possible that we track objects without ever regarding their color or shape. More likely is that color and shape come into play when tracking based on velocity fails. It also seems likely that shape and color are more closely tied to one's

world knowledge and might be used as “hooks” to relate things we know about a scene to the objects we’re seeing in that scene.

“Object perception does accord with principles governing the motions of material bodies: Infants divide perceptual arrays into units that move as connected wholes, that move separately from one another, that tend to maintain their size and shape over motion, and that tend to act upon each other only on contact.” [32]

The literature, therefore, points to a hierarchy of features that are utilized when humans perform tracking:

1. Velocity - at the lowest level, objects are delineated and tracked based on their simultaneous movement. Things that tend to move together tend to be single objects.
2. Color - areas of the visual array that exhibit coherent color through temporal and spatial change tend to be classified as objects.
3. Shape - this is the most complex feature to understand as it involves complex integration with world knowledge due to the geometric variability of many objects. A person’s shape, for example, changes dramatically over time, but we are still able to recognize this multitude of different shapes as a person. Even given the problems and complexities associated with shape-based tracking, shape appears to be a feature that is utilized in the human tracking system, and one that has also proven useful in machine vision.

Implementation

2C was developed around 3 primary datasets. In all cases, video was generated by a network of overhead cameras with fisheye-style lenses. The primary dataset was the Human Speechome Project corpus, with other datasets collected from inside busy retail environments. Properties such as average number of people, variety of lighting, and motion patterns of people vary enormously between datasets, making them ideal for development of a general tracking system. All video is 960 pixels x 960 pixels and is encoded in a proprietary format based on motion-JPEG.

2C is written primarily in Java, with certain aspects written in C and accessed from Java using JNI. The software consists of approximately 20,000 lines of code altogether.

2C implements a pipeline architecture. First, the pipeline is defined in terms of the various modules that will make it up. An input component accepts digital video (various video formats are currently supported). This input is then passed sequentially to each module in the pipeline, along with a data structure that carries the results of any processing a module undertakes. An output module operates on this data structure, producing whatever output is desired. Modules can be defined to perform any arbitrary operation on either the input or the output of modules that come before it in the pipeline. In this way, dependencies can be created such that modules work together to perform complex functions. Pipelines can be defined and modified on the fly, making it possible to implement a dynamic system where various modules are activated and deactivated regularly during processing. To date, modules exist to handle nearly any video input format, to perform image processing and analysis tasks including various types of feature extraction (such as color histogram generation and SIFT feature [15] generation), and to produce output of various kinds, including numerical and visualization.

2.2.3 Input Component

The input component decodes a proprietary video format based on motion-JPEG known as “squint” video. Each frame of variable framerate video contains a microsecond-accurate timestamp. A key design choice implemented in this component is the decision to utilize partially decoded video frames (known as “wink” video). This results in 120x120 frames, as opposed to 960x960, speeding processing considerably through the entire pipeline, particularly in the input and background subtraction phases.

2.2.4 Low-level Feature Extraction

Tracking begins with motion detection and clustering. These processes form the low-level portion of the system and can be thought of as an attention allocation mechanism. Many biologically plausible attention mechanisms have been proposed [21, 24] and likewise many computational algorithms have been developed [33, 29, 16], all with the aim of segmenting a scene over time into “background” and “foreground” with foreground meaning areas of a scene that are salient, as opposed to areas that are physically close to the viewer. Areas that are considered foreground are then further segmented into discrete objects. These objects can then be tracked from frame to frame by higher level processes.

Motion detection

The motion detection process operates using a frame-differencing operation, where each pixel of each new frame of video is compared to a statistical model of the background. Pixels that do not appear to be background according to this comparison are classified as foreground. The background model is then updated using the new frame as a parameter.

The output of the motion detection step is a binary image D where each pixel $d_i = 0$ indicates that d_i is a background pixel and $d_i = 1$ indicates that d_i is foreground.

The algorithm implemented is a mixture-of-gaussians model as described in [33], where each pixel X_i 's observed values are modeled as a mixture of K gaussians in 3 dimensions (RGB) $Q_i = q_0 \dots q_K$, each with a weight w_k . Weights are normalized such that $\sum_k^K w_k = 1$. A model Q_i is initialized for each pixel i , then each new frame is compared to this model such that each pixel is either matched to an existing gaussian or, when the new pixel fails to find a match a new gaussian is initialized. Newly initialized gaussians are given a low weight. Matches are defined as a pixel value within some multiple of standard deviations from the distribution. In practice, this multiple is set to 3.5, but can be adjusted with little effect on performance.

Each pixel is therefore assigned a weight w_i corresponding to the weight of its matching (possibly new) distribution. We can then classify each pixel according to a parameter T denoting the percentage of gaussians to consider as background:

$$d_i = \begin{cases} 0 & : w_i > T \\ 1 & : otherwise \end{cases}$$

Weights are then adjusted according to a learning parameter α corresponding to the speed at which the model assimilates new pixel values into the background:

$$w_{j,t} = (1 - \alpha)w_{j,t-1} + \alpha(M_{k,t})$$

where $M_{k,t} = 1$ for the matching distribution and is 0 otherwise. Weights are normalized again so that $\sum_k^K w_k = 1$.

Parameters for the matching distribution are adjusted as follows:

$$\mu_t = (1 - \rho)\mu_{t-1} + \rho X_t$$

$$\sigma_t^2 = (1 - \rho)\sigma_{t-1}^2 + \rho(X_t - \mu_t)^T(X_t - \mu_t)$$

where:

$$\rho = \alpha\eta(X_t|\mu_k, \sigma_k)$$

This model has several advantages. First, it is capable of modeling periodic fluctuations in the background such as might be caused by a flickering light or a moving tree branch. Second, when a pixel is classified as background, it does not destroy the existing background model - existing distributions are maintained in the background model even as new distributions are added. If an object is allowed to become part of the background and then moves away, the pixel information from before the object's arrival still exists and is quickly re-incorporated into the background.

The input to the motion detection process is raw visual field data, and the output consists of pixel-level motion detections, known as a difference image.

Motion clustering

Foreground pixels are grouped into larger detections by a motion clustering process. This process looks for dense patches of motion in the set of detections produced by the motion detection process and from those patches produces larger detections consisting of size, shape, and location features.

This module iterates over patches in the difference image produced above and computes a density for each patch where density is the number of white pixels / the total number of pixels in the patch. Patches with density greater than a threshold are then clustered to produce larger areas representing adjacent dense areas in the difference image. These larger dense areas are called particles.

Pseudocode for this algorithm follows:

```
foreach ( $n \times n$ ) patch in difference image do
  if patch(white) / patch(total) > threshold then
    | add patch to patchList
  end
end
foreach patch in patchList do
  foreach existing particle  $p_i$  do
    | if intersects(patch,  $p_i$ ) then
      | | add(patch,  $p_i$ )
    | end
    | new( $p_j$ )
  end
end
```

The input to the clustering process is the pixel-level detections output by the motion detection process, while the output is larger aggregate motion detections.

2.2.5 Object tracking

Once low-level features are extracted from the input, the visual system can begin segmenting objects and tracking them over time.

Motion-based hypothesis

Based on the detections provided by the motion clustering process, the motion tracking algorithm computes spatio-temporal similarities and hypothesizes the locations of objects in the visual field. In other words, it makes guesses as to where things are in the scene based on the motion clustering process's output. It does so by computing velocities for each object being tracked, and then comparing the locations of detections to the expected locations of objects based on these computed velocities. Detections that share coherent velocities are therefore grouped into objects, and those objects are tracked from frame to frame (see Figure 2-4).

Classifiers

The motion tracking algorithm makes a binary decision as to whether to associate a particle p_i with an existing object o_j . These decisions are made on the basis of either an ad-hoc heuristic classifier, or a learning-based classifier trained on ground truth track data.

1. Heuristic classifier - this classifier attempts to embody the kinds of features a human might look for when making decisions. It works by computing an association score, and then comparing that score to a threshold in order to make its decision. The parameters of this classifier (including the threshold) can be tuned manually (by simply watching the operation of the tracker and adjusting parameters accordingly) or automatically using gradient descent on a cost function similar to the MOT metrics described below.

The association score is computed as follows:

$$\Theta(p_i, o_j) = \alpha_1(\Delta_v(p_i, o_j)) + \alpha_2(\Delta_d(p_i, o_j))$$

Δ_v is the difference in velocity of p_i (assuming p_i is part of o_j) and o_j before connecting p_i .

Δ_d is the Euclidean distance between p_i and o_j .

α_1 and α_2 are gaussian functions with tunable parameters.

2. Learning-based classifier - these classifiers use standard machine learning techniques in order to perform classification. Ground truth tracks are generated using a human annotator. These tracks can then be used to train the tracker's output, with positive and negative examples of each classification task being generated in the process. Classifiers that have been tested include Naive Bayes, Gaussian Mixture Models, and Support Vector Machines. All perform at least moderately well, with certain classifiers exhibiting particular strengths. In practice, however, the heuristic classifier described above is used exclusively.

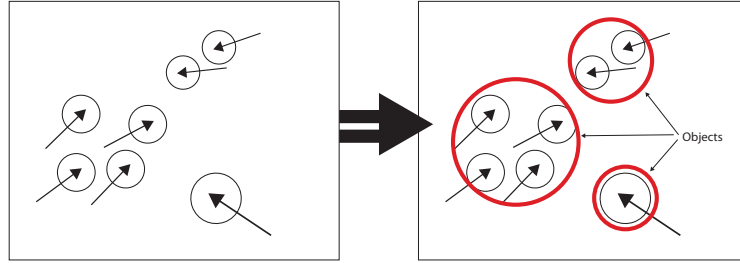


Figure 2-4: Motion-based tracking. Detections are clustered into objects that share coherent velocities.

The motion tracking algorithm exhibits several useful properties. One such property is the tracker's ability to deal with noisy detections. If an object is split across several detections (as often happens), the tracker is able to associate all of those detections to a single object because their velocities are coherent with that object. Likewise if several objects share a single detection, that detection can provide evidence for all objects that exhibit coherent

motion with that detection.

Motion tracking also encodes the fundamental notion of object permanence. Once an object has been built up over time through the observation of detections, the tracker maintains that object in memory for some period of time, looking for further detections that support its location. This notion of object permanence also helps the tracker deal with errors in motion detection - a common problem in motion-based tracking is maintaining object location when that object stops moving. Here we maintain the object's location even without evidence and then resume normal tracking when the object finally moves again and new evidence is provided.

This module accepts the set of clustered motion detections (particles) produced above as input and attempts to infer the locations of objects. It does so by making an association decision for each particle/object pair. If a particle is not associated with any existing object, a decision is made whether to instantiate a new object using that particle. If a new object is not instantiated, the particle is ignored (treated as noise).

Color-based hypothesis

In each frame, color-based tracking is performed in addition to motion-based tracking. For a given object, we perform Meanshift [3] tracking in order to formulate a hypothesis as to that object's position in the new frame. This algorithm essentially searches the area immediately around the object's previous known position for a set of pixels whose colors correspond to the object's color distribution.

Meanshift works by searching for a local mode in the probability density distribution representing the per-pixel likelihood that that pixel came from the object's color distribution. Color distributions are aggregated over the lifespan of each object, and are updated periodically with color information from the current video frame. Color distributions are represented as 3-dimensional (red, green, blue) histograms.

Mean shift follows these steps:

1. Choose an initial window size and location
2. Compute the mean location in the search window
3. Center the search window at the location computed in Step 2
4. Repeat Steps 2 and 3 until the mean location moves less than a preset threshold

The search window size and position W are chosen as a function of the object's location at time $t - 1$: $W_t = \theta(O_{t-1})$. Call the object's current aggregate color distribution Q_t . We first compute a probability image I as: $I(x, y) = Pr((x, y); Q_t)$ or the probability that pixel (x, y) comes from distribution Q_t for all values of (x, y) in the current frame of video. We then compute a the mean location $M = (\tilde{x}, \tilde{y})$ as:

$$\tilde{x} = \frac{\sum_x \sum_y x I(x, y)}{\sum_x \sum_y I(x, y)} \quad \tilde{y} = \frac{\sum_x \sum_y y I(x, y)}{\sum_x \sum_y I(x, y)}$$

This process continues until M moves less than some threshold in an iteration. In practice, M generally converges in under 5 iterations.

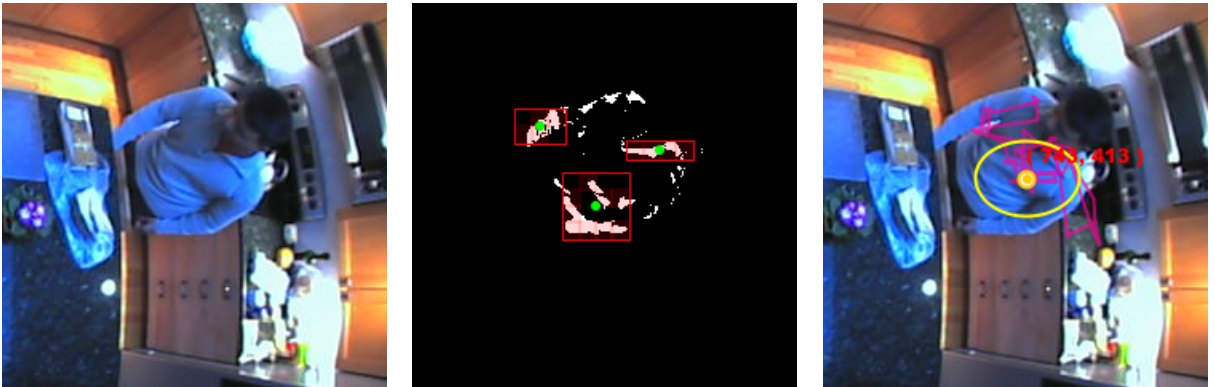


Figure 2-5: Tracking pipeline: raw video, motion detection and aggregation, tracking

Hypothesis integration

For each object at time t , we have a motion hypothesis \hat{O}_t and a color hypothesis \tilde{O}_t . These hypotheses are combined using a mixing parameter α such that the overall hypothesis $O_t = \alpha\hat{O}_t + (1 - \alpha)\tilde{O}_t$.

Hypothesis revision

This step searches for objects that should be merged into a single object, or those detections that were incorrectly tracked as two or more objects when they should have been part of a single object. In order to make this determination, pairwise merge scores $S_{i,j}$ are generated for all objects:

$$S_{i,j} = \sum_{n=0}^N \psi_n \Theta(O_{i,t-n}, O_{j,t-n})$$

where:

$\Theta(O_{i,t-n}, O_{j,t-n})$ is the association score from above and:

ψ_n is a weighting parameter denoting how much more weight to place on more recent observations.

This score therefore denotes the average likelihood that objects i and j are associated (are the same object) over N steps back from the current time t . When $S_{i,j} > T$ where T is a merge threshold, we merge objects i and j .

Object de-instantiaion

When motion tracking fails to provide evidence for an object, we look to the color distribution to determine whether to de-instantiate the object. This says, in effect, that if we have no motion evidence (the object has come to rest), but the colors at the object's last known location match closely to the object's aggregate color distribution, then we maintain our hypothesis about that object's location. However, if the colors do not match, we de-instantiate that object. We are thus making a binary decision whenever we lose motion

evidence for an object, where 0 = remove the object and 1 = maintain the object hypothesis. In practice, we allow a window without motion evidence proportional to an object's lifespan *with* motion evidence before we force the system to make its color-based binary decision.

We compute this binary decision as follows:

First compute the Bhattacharyya Distance $D_B(P_t, Q_t)$ where P_t is the pixel color distribution at the object's current location and Q_t is the object's aggregate color distribution taken at time t : $D_B(P, Q) = -\ln(BC(P, Q))$

where:

$$BC(P, Q) = \sum_{x \in X} \sqrt{p(x)q(x)}$$

is the Bhattacharyya Coefficient and X is the set of pixels.

The decision $K(O_i, t) \in \{0, 1\}$ whether to de-instantiate object O_i at time t with threshold T is then:

$$K(O_i, t) = \begin{cases} 0 & : D_B(P_t, Q_t) > T \\ 1 & : otherwise \end{cases}$$

Algorithm summary

Given the set of particles $P_t = \{p_0, \dots, p_n\}$ at time t and the current set of objects $O_t = \{o_0, \dots, o_k\}$, the algorithm is summarized as follows:

```

foreach  $p_i$  do
|   foreach  $o_j$  do
|   |    $\text{associate}(p_i, o_j)$ 
|   end
end

foreach  $p_i$  with no associated  $o_j$  do
|   instantiate new o
end

foreach  $o_j$  do
|   perform meanshift tracking
end

integrate motion and color hypotheses

foreach  $o_j$  with no associated  $p_i$  do
|   de-instantiate  $o_j$ ?
end

foreach  $o_j$  do
|   merge  $o_j$  with other objects?
end

```

Algorithm 1: Tracking algorithm

2.2.6 Performance

2C is evaluated along two dimensions. First, we look at standard accuracy and precision measures to evaluate the quality of the output of the system. Second, the speed at which 2C is able to generate those results is taken into consideration.

MOT Metrics

In order to be able to evaluate the tracking system's performance, we need a robust set of metrics that is able to represent the kinds of errors that we care about optimizing. One such set of metrics are the Clear MOT metrics, MOTA and MOTP (Multiple Object Tracking Accuracy and Multiple Object Tracking Precision) [1]. In this work, I use a modified MOTA

and MOTP score that reflect the need to find accurate points while ignoring the contiguity of tracks in favor of increased efficiency.

To compute MOT metrics, we first produce a set of ground truth tracks via manual annotation. Several such annotation tools have been developed, the most basic of which simply displays a video sequence and allows the user to follow objects with the mouse. More sophisticated versions incorporate tools for scrubbing forward and backward through video, tools for stabilizing tracks, tools for automatically drawing portions of tracks, etc. Ground truth for this work was produced primarily via two tools: Trackmarks [5] and a lightweight, custom Java application that produces ground truth track data by following the mouse’s movement around the screen as the user follows a target in a video sequence.

MOTA and MOTP are computed as follows:

Given the set of ground truth tracks and a set of hypothesis tracks that we wish to evaluate, we iterate over timesteps, enumerating all ground truth and hypothesis objects and their locations at each time.

At time 0 initialize an error count $E = 0$ and a match count $M = 0$

We then create the best mapping from hypothesis objects to ground truth objects using Munkres’ algorithm [38], and then score this mapping as follows:

For each correct match, store the distance d_t^i , increment $M = M + 1$ and continue.

For each candidate for which no ground truth object exists (false positive), increment $E = E + 1$

For each ground truth object for which no hypothesis exists (miss), increment $E = E + 1$

MOTP is then:

$\frac{\sum_{i,t} d_t^i}{M}$ or the distance error averaged over all correct matches.

MOTA is:

$\frac{E}{E+M}$ or the ratio of errors to all objects.

Table 2.1 shows MOTA and MOTP scores as well as average track duration for 2C, as well

	2C	SwisTrack
MOTA	0.74	0.48
MOTP	1,856.14	2043.47
totalTimesteps	3,479	3,479
totalObjects	11,896	11,896
totalHypotheses	9,795	7,426
totalMatches	9,294	6,606
totalFalsePositives	501	820
totalMisses	2,602	5,293
totalMistakes	3,103	6,113
Mean track duration (sec)	56.8	13.9

Table 2.1: Accuracy and precision comparison

as for SwisTrack [14], an open source vision architecture that has previously been applied to HSP data and that serves as a useful baseline for tracking performance.

The interpretation of these scores is that 2C is approximately 74% accurate, and is precise to within 1.8m on average. Further inspection of the statistics reveal that misses (cases where there is an object that the tracker fails to notice) are more than 5 times more common than false positives (when the tracker denotes the presence of a non-existent object). In our application, this is an acceptable ratio, as misses damage the results very little while false positives have the potential to corrupt findings far more. Although it was not a primary consideration in its design, notice that 2C produces longer tracks than SwisTrack (56.8 sec vs. 13.9 sec), which is particularly encouraging in light of 2C’s substantially higher MOTA and MOTP scores (notice that due to the near complete recording coverage of the home, we can assume that “correct” tracks will often be long, breaking only when a subject either leaves the home or enters an area without video coverage).

Speed

Speed of processing was a primary consideration in the design and implementation of 2C. As such, real world processing speed was analyzed and tuned exhaustively. Evaluations given here are for a single process running on a single core, however in practice 2C was run in an environment with many computers, each with up to 16 cores, all running in parallel.

	Mean	Std
Input Component	< 1	< 1
Background Subtraction	2.1	3.32
Motion Aggregation	< 1	.03
Tracker	0.43	6.08
Output Component	< 1	2.3
Total frame time	4.10	6.81

Table 2.2: Runtime stats for tracking components

	Mean	Std
Init	0.03	2.38
Matching	0.1	0.39
Color Tracking	0.24	0.48
Integrate Hypotheses	0.01	0.12
Merge	0.05	2.26
Prepare Output	0.01	0.37

Table 2.3: Runtime stats for tracking module steps

Per-core speeds were slower, but overall throughput was of course much faster.

Runtime for each component is given in Table 2.2 and a breakdown by each step in the tracking algorithm is provided in Table 2.3 (all times are in milliseconds). Precise runtime data is unavailable for SwisTrack, but observed speeds across many tracking tasks was near real time (67ms/frame for 15fps video).

2.2.7 Tuning

An effort was made to control the free parameters in the 2C system in two ways. First, I attempted simply to minimize the number of free parameters. This was done by simplifying where possible, combining parameters in sensible ways, and allowing the system the freedom to learn online from data whenever possible. This effort was balanced against the desire to “bake in” as little knowledge of tracking as possible, requiring the abstraction of many aspects of the operation of the system out into new free parameters.

The second part of the effort to control 2C’s free parameters involved the framing of the purpose of these parameters. Rather than allowing them to be simply a set of model parameters for which no intuitive meaning is possible, the free parameters are all descriptive in terms that are understood by a human operator of the system. For example, consider the set of parameters used in performing association of particles to objects. These have names such as “WEIGHT_DISTANCE”, “WEIGHT_VELOCITY”, and “MIN_ASSOCIATION_SCORE” with intuitive explanations such as “the weight to apply to the Euclidean distance score between particle and object when computing the overall score” and “the minimum overall score for which an association is possible.” Contrast this to a more abstract tracking approach such as a particle filter based tracker, where there is a set of parameters for which no human-friendly description is possible.

Even with the parameter list minimized, the search space for parameter settings is large. For this reason, two methodologies have been explored and utilized for establishing optimal values for the free parameters in the 2C system. First, a GUI was created that allows the user to manually change the various parameter settings while watching an online visualization of the tracker’s operation. This method heavily leverages the human operator’s insights about how to improve tracker performance. For example, a human operator might realize that the operation of the motion tracking algorithm is highly sensitive to the output of the background subtraction algorithm, and might choose to tune background subtraction while “keeping in mind” properties of motion tracking. This allows the human operator to traverse locally poor settings in pursuit of globally optimal ones.

The second approach to tuning free parameters is an automatic one and uses a gradient descent algorithm. A set of target parameters to tune is defined, as well as an order in which to examine each parameter and default values for the parameters. Then, with all other parameters held constant at their default values, the tracker is run iteratively with all possible values of the initial target parameter. The best value of these is chosen, and that value is then held constant for the remainder of the optimization run. Values for the

next parameter are then enumerated and tested, and so on until all parameters have been set to optimal values. We then begin another iteration, resetting all parameter values. This process continues until parameters are changed less than some threshold in a given iteration. This method tends to find good values for parameters, but suffers from local maxima and is highly sensitive to both the initial values of parameters and the definition of the tuning set and order.

A variation of the second approach utilizes a genetic algorithm in an attempt to more fully explore the parameter space. Initial values are set at random for all parameters. Gradient descent then proceeds as above until all values have been reset from their random starting points. This final set of parameters is saved, and a new set of initial values is set at random. The process proceeds for n_0 steps, when the overall best set of parameters is chosen from among the best at each step. This overall winner is then perturbed with random noise to generate n_1 new sets of starting values. Each of these starting value sets is optimized using gradient descent as before, again with the overall best optimized set being chosen. This process proceeds for k iterations. This method more fully explores the search space, but is extremely computationally expensive. For example, if we are tuning r parameters and enumerate m possible values for each, then we must track $(n_0 + \dots n_k) * (m * r)$ video sequences. This number grows large quickly, particularly if we are tracking full-resolution video in real time. Tuning 10 parameters with 10 values each with 5 initial random sets at each iteration for 5 iterations with a 5 minute video sequence results in a total runtime of 12,500 minutes (208 hours).

While all three approaches described above were tested, the best results came from a combination of manual and automatic tuning. Initial values were set manually via the GUI. These values were used as starting points for several iterations of gradient descent. The final values from gradient descent were then further optimized manually, again using the GUI.

2.3 Transcription and Speaker ID

Audio data is transcribed via a semi-automatic system called BlitzScribe [27]. BlitzScribe works by first segmenting the audio stream into discrete utterances. Segmentation is done by searching for silence, and then by optimizing utterance length based on the cuts proposed by the silence. Utterances are then aligned with human annotation of the location of the child such that only utterances representing “child-available” speech are marked for transcription. Audio is then given to transcribers one utterance at a time to be transcribed. Transcribed segments are stored as text in an encrypted SQL database, each with start and stop times (in microseconds), the audio channel from which the utterance originated, and the annotation of the child’s location. To date, approximately 60% of the corpus has been transcribed.

Speaker identity is determined automatically using a generative model-based classification system called WhoDat [18]. In addition to identity, WhoDat produces a confidence score denoting its certainty about the label it has attached to an audio segment. Identity is added to each utterance in the database along with transcripts.

Transcription accuracy is checked regularly using a system of inter-transcriber agreement, whereby individual transcripts may be marked as inaccurate, or a transcriber’s overall performance can be assessed. Speaker ID was evaluated using standard cross validation techniques. Performance varies considerably by speaker, with a high accuracy of 0.9 for the child and a low of 0.72 for the mother, using all utterances. If we assess only utterances with high confidence labels, accuracy improves significantly, at the expense of the exclusion of substantial amounts of data. In practice, a confidence threshold of 0.4 is used when speaker identity is needed (such as when determining which utterances were made by the child), resulting in over 90% accuracy across all speakers and yielding approximately 2/3 of the data.

2.4 Processed Data

Tracks generated by 2C and transcripts (with speaker ID) from BlitzScribe are then further processed to derive the datatypes described below.

2.4.1 Processed Tracks

Tracks are projected from the pixel space of the video data where it was recorded into world space, represented by Euclidean coordinates relative to a floorplan of the home. The fisheye lenses of our cameras are modeled as spheres, and model parameters θ are derived using a manual annotation tool. θ fully specifies the camera's position and orientation in world space. Each point P in a given track can then be mapped to world space U by a mapping function $f(P : \theta) \rightarrow U$.

Once projected into the single coordinate system representing the entire home, tracks can be aggregated across all cameras. These aggregate tracks are Kalman filtered [10] and point reduced using the Douglas-Peucker algorithm [39]. Once aggregated and filtered, tracks are

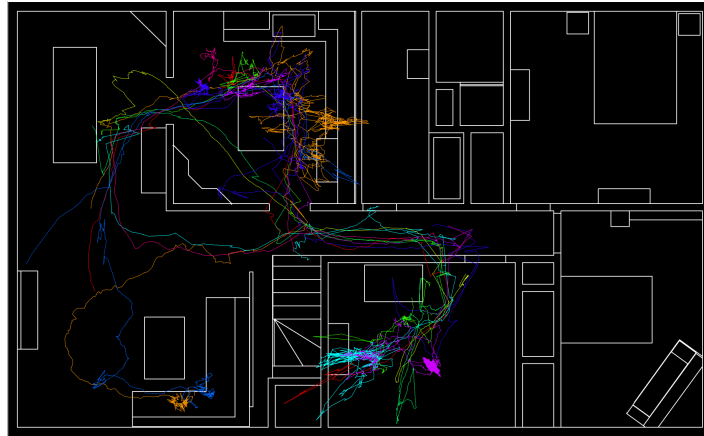


Figure 2-6: Sample Movement Traces

merged across cameras. This process attempts to join tracks from adjacent cameras that represent the same tracked subject. Merging proceeds as follows. For two sets of tracks in adjacent cameras, we generate all pairwise scores between individual tracks. The score is computed as the mean distance between temporally overlapping portions of the two tracks, combined with the point-wise standard deviation between the tracks in a weighted average. This formulation incorporates two assumptions about tracks that should be merged: that they should be close together for their duration (low mean delta distance), and that regardless of their distance, they should maintain a somewhat constant distance from each other (low standard deviation).

The score $S_{i,j}$ for tracks i, j is computed as:

$$S_{i,j} = (\beta_1 d_{\hat{i}, \hat{j}}) + \beta_2 \sigma_{\hat{i}, \hat{j}}$$

where:

$d_{\hat{i}, \hat{j}}$ = mean distance between tracks \hat{i}, \hat{j}

β_1 and β_2 are tuned parameters

and:

\hat{i}, \hat{j} are the portions of track i and track j that overlap in time.

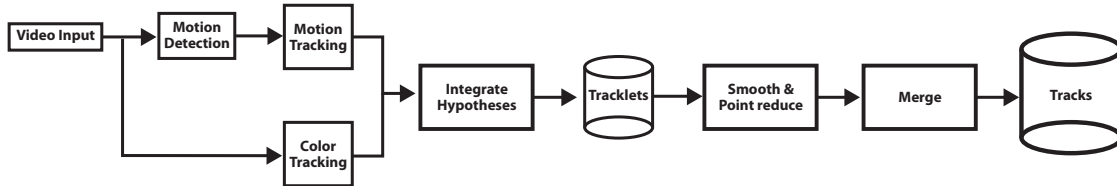


Figure 2-7: Track processing pipeline

Each track is then iteratively merged with all other tracks whose score is below a threshold. This threshold was tuned empirically by iterating over values and examining both visualizations of the resulting merged tracks, as well as raw video data corresponding to the objects being tracked.

The output of the track processing step is a set of tracks corresponding to all movement

throughout the home during the period of recording. These tracks are stored in SQLite database files, with one day per file.

2.4.2 Child’s Vocabulary and Word Births

From the transcripts of audio data, we’d like to know which words were present in the child’s vocabulary by the age of 2, as, by definition, these are the individual words that signify language acquisition in the child. Then, for each of these words we would further like to know the time of that word’s first production. Given perfect transcription and speaker ID, this is a trivial process, easily handled by a single query to the database (i.e. `SELECT * FROM utterances WHERE timestamp == min(timestamp) AND speaker == “child”`). Both transcription and speaker ID are imperfect, however, which necessitates some filtering in order to find first the child’s vocabulary and then the first production of each word in the vocabulary.

First I generated the vocabulary for the entire Human Speechome corpus by iterating over all transcription and storing unique tokens. This resulted in 24,723 unique tokens, with 1,772 having appeared more than 100 times. To mark a word as part of the child’s vocabulary, it must appear a minimum of 10 times throughout the corpus, marked as “[child]” with high confidence by speaker ID. This list is then filtered to remove non-linguistic tokens, as well as to manually map various forms of the same word to a single token (for example “dad,” “daddy,” and “dada”). This process resulted in 658 words being identified as present in the child’s vocabulary (see Appendix A).

In order to establish the time of the child’s first production of each word or Age of Acquisition (AoA), I create per-word temporal distributions at the week and month timescales. I then search for the knee in each distribution, or the point at which the child’s use of the word increases substantially. This step helps to avoid spurious false positives before the child actually assimilated a word into his vocabulary. The knee at each timescale is averaged. Given this average knee, we then search for the nearest production of the word by the

child and call this the word birth, with its timestamp being that word’s AoA. These timestamps are more accurately denoting the age at which the child first assimilates a word into his vocabulary; however, this is assumed to be closely related to the time of first production and so is used as the age of acquisition time.

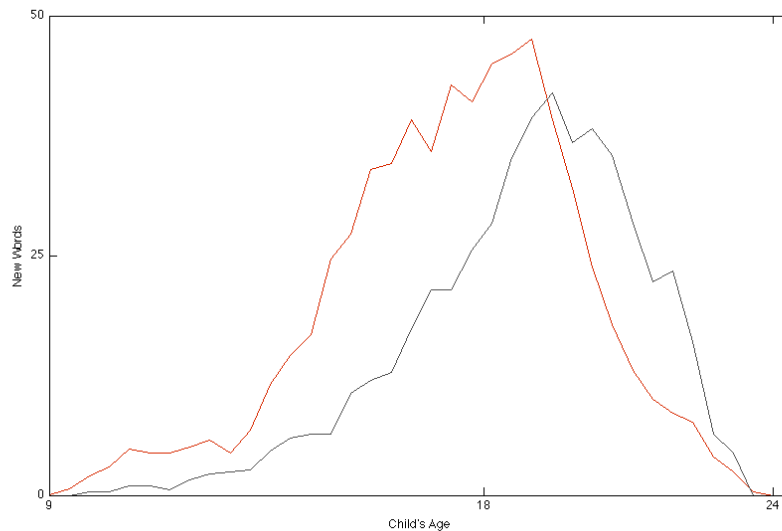


Figure 2-8: Old vs. New Word Births

As a check on the results of this step, I gathered Age of Acquisition data derived for previous research. This data was derived when there was substantially less transcription complete, so we might expect AoA to move forward in time as we see new child-spoken utterances containing a given word. Figure 2-8 shows that this is in fact the case - the overall pattern of word acquisition (the “shark’s fin”) remains nearly identical, while the timestamps for each word move forward in time in almost all cases. As another check on the newly derived age of acquisition for each word, I plotted the child utterance temporal distributions, along with the newly derived and previous word birth timestamps (see example in Figure 2-9). These simple plots convey information about the child’s usage of a word, and proved powerful in troubleshooting AoA data. As a final check on each AoA, transcripts were examined for each word birth utterance. In several cases, reading the transcript showed that an utterance couldn’t have been produced by the child, necessitating manual intervention to find the true

first production of the word by the child.

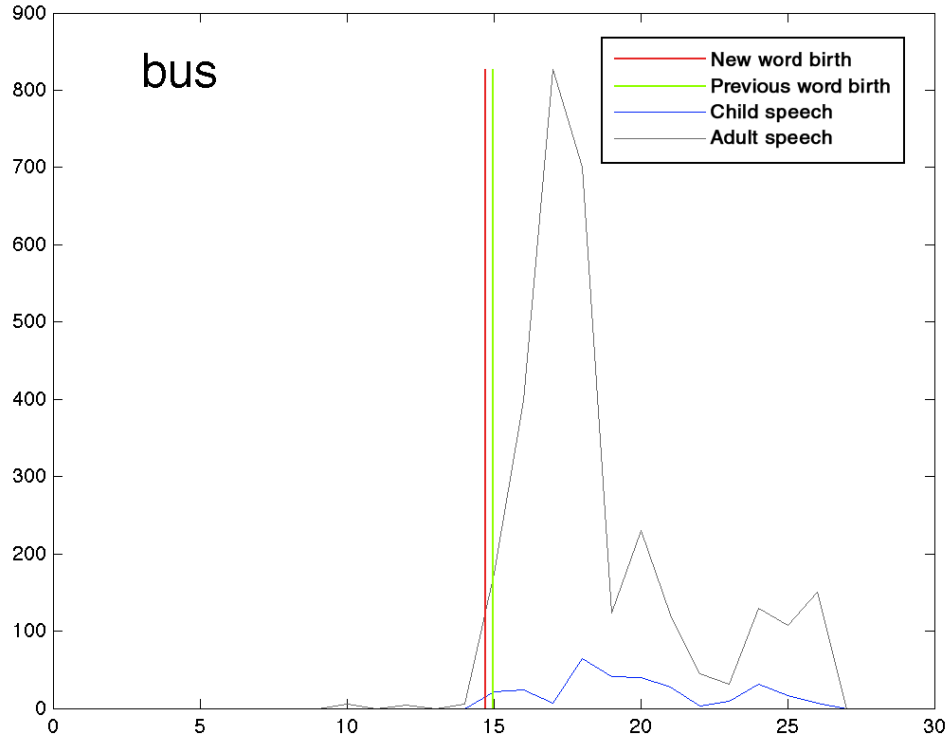


Figure 2-9: Word birth verification plot

2.4.3 Situated Utterances

For a given utterance, I attempt to “situate” that utterance by extracting the point or set of points denoting the location of a person or people at the time of the utterance. To do this, I search for all tracks whose start and end times intersect the start and end times of the utterance, and then extract (or interpolate) one point from each intersecting track at the timestamp of the midpoint of the utterance. These points are then stored in a table, matched to the target utterance. The result of this step is a table that stores the location of participants for each utterance in the corpus.

2.4.4 Spatial Distributions

Situated utterances are distilled into spatial histograms that represent aggregate views across arbitrary dimensions. A 2-dimensional histogram is initialized where the bins correspond to discrete locations in the home. Histograms are initialized for bin sizes of 100mm and 1000mm, with bins distributed in a uniform grid throughout the space. For 100mm bin sizes, distributions contain $162 \times 118 = 19,116$ bins. The 1000mm distributions contain $16 \times 11 = 176$ bins.

For each situated utterance of interest, the set of points corresponding to the location of people at the time of that utterance are added to the appropriate bin(s) of the histogram using bilinear interpolation. Each bin is given a weight corresponding to the area an artificial bin centered at the point would overlap with the bin in question. A weighted point is then added to each bin. Note that by this method, at most 4 bins can be affected by a single point and a point that falls directly in the center of a bin affects only that bin.

If we have a point P and a bin centered at K with size $w * h$, the weight $\Gamma_{P,K}$ is given by:

$$\begin{aligned}
P_x^1 &= (P_x - w/2) & P_x^2 &= (P_x + w/2) \\
P_y^1 &= (P_y - h/2) & P_y^2 &= (P_y + h/2) \\
K_x^1 &= (K_x - w/2) & K_x^2 &= (K_x + w/2) \\
K_y^1 &= (K_y - h/2) & K_y^2 &= (K_y + h/2) \\
\Gamma_{P,K} &= \frac{[\min(K_x^2, P_x^2) - \max(K_x^1, P_x^1)] * [\min(K_y^2, P_y^2) - \max(K_y^1, P_y^1)]}{w * h}
\end{aligned}$$

This spatial distribution represents the aggregate locations of participants in the utterances of interest. Histograms are represented as multinomials with the added property that bins have spatial adjacencies, where k = the number of bins and n = the number of samples (in this case utterance points). The probability of an utterance occurring at a location i is the total count of points in $i = X_i$ divided by the total number of points n : $p_i = \frac{X_i}{n}$ and

$\sum_i^k p_i = 1$. The mean location is a weighted sum of bin locations, where p_i is the weight of location i and K_i is the coordinate: $\mu = \sum_i^k p_i K_i$ and the mode is simply the maximum likelihood location: $mode = K_i \text{ s.t. } i = \operatorname{argmax}(p_i)$.

2.5 Summary

Figure 2-10 summarizes the dataset creation pipeline. Tracks are produced by 2C, then are filtered and merged across cameras. Transcription created by BlitzScribe is used to generate the child’s vocabulary and word birth dates. Processed tracks and transcription are then joined to form situated utterances. These are aggregated to form spatial histograms.

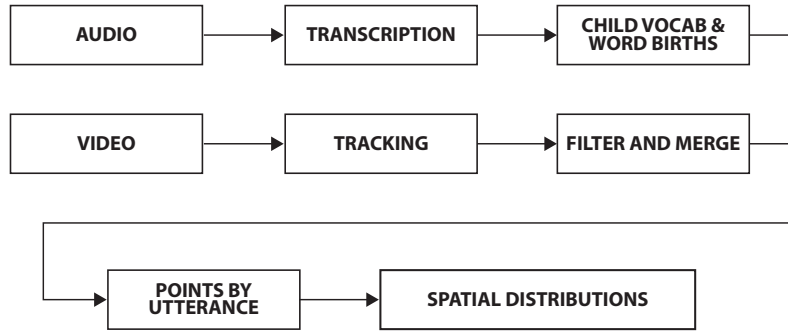


Figure 2-10: Summary of Dataset Processing

Chapter 3

Taxonomy

3.1 Overview

The fundamental building blocks of the taxonomy described here are spatial distributions representing the locations of people during normal daily life. These distributions carry with them various metadata, including the speech type (i.e. a particular word) they represent and various statistical measures that serve to quantify the distribution. Distributions are visualized in several ways for presentation to the user.

3.2 Schema

The schema for the taxonomy is defined according to a 3-dimensional structure as follows. Each axis is segmented by a dimension of interest: activity type, participant identity, and temporal slice. Locations along all axes are discrete.

Along the y-axis, we have activity types. With the exception of the first entry, activity is speech and is defined according to the content of the speech.

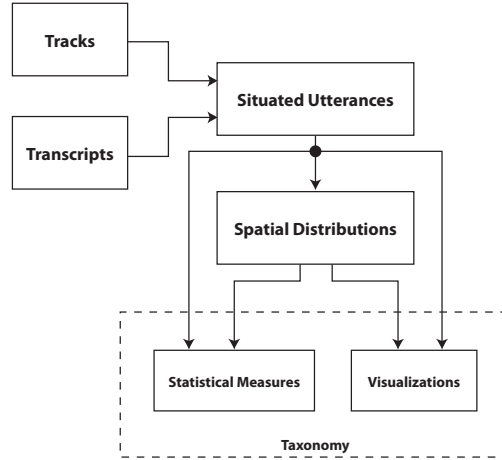


Figure 3-1: Overview of Dataset and Taxonomy

Entries on the y-axis are:

Activity: This represents all person tracks in the corpus (note that identification is currently done only on the basis of speech, therefore activity entries are not segmented by identity)

Speech: This represents data for all speech acts in the corpus

Target Words: These are utterances containing any of the 658 words in the child’s vocabulary at age 2

Learning Period: For each of the target words, these are utterances containing that word that occurred before the child’s first production of the word.

Target Words and the Learning Period are further segmented by each of the individual words.

The y-axis therefore contains $(1 + 1 + 1 + 1 + 658 + 658) = 1,320$ entries.

Along the x-axis, we have participant identities. These identities are segmented as follows:

All participants: no filtering is done

Mother: only utterances made by the Mother are included

Father: only utterances made by the Father are included

Nanny: only utterances made by the Nanny are included

Child: only utterances made by the Child are included

Other: utterances made by participants other than those noted above are included

The x-axis contains 6 entries.

To determine the total number of entries across the x- and y-axes, we first note that identity is not available for non-speech activity traces because identity is derived from utterance audio. We also note that, by definition, Learning Period utterances are not made by the child, so these entries are empty and need not be counted. We can now determine the total number of entries as: $(6 * (1 + 1 + 658)) + (5 * (1 + 658)) + 1 = 7,256$

Along the z-axis, we place temporal slices. While temporality can be viewed continuously, we instead discretize as follows:

All: all activity

Morning: activity taking place between 4am and 9am

Daytime: activity taking place between 9am and 5pm

Evening: activity taking place between 5pm and 8pm

Night: activity taking place between 8pm and 4am

Weekend: activity taking place on Saturday or Sunday

Weekday: activity taking place Monday - Friday

By month: activity corresponding to a single month in the child's life from 9 - 24 months.

Combined with the entries above, the complete taxonomy contains $7,256 * (7+16) = 166,888$ entries altogether.

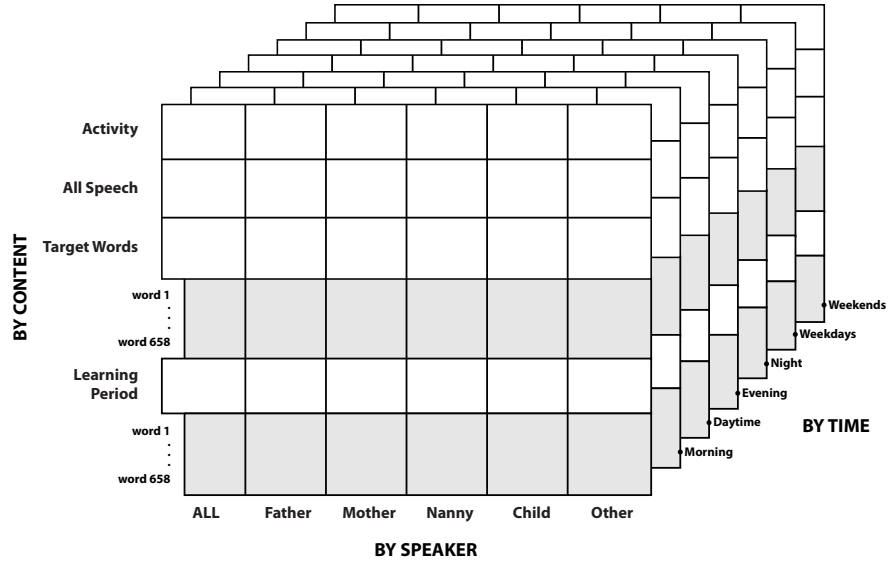


Figure 3-2: Taxonomy schema

3.3 Visualizations

For each entry in the taxonomy, the following visualizations were produced (details about each type follow):

- Heat map (standard) for 100mm and 1000mm bin size distributions
- Heat map (log scale) for 100mm and 1000mm bin size distributions
- Difference map comparing this entry to the other entries in its x, y, and z axes (i.e. target word utterances made by the father compared to all target word utterances) for 100mm and 1000mm bin size distributions

3.3.1 Heat Maps

The core visualization type represented in the taxonomy are heat maps utilizing a “rainbow” spectrum of color to represent counts in the various bins. These heat maps are normalized such that the maximum value is depicted in white and the minimum value is black. These basic heat maps are also extended to heat maps plotted on a log scale, again normalized so that the maximum is white and the minimum is black. The log scale versions are useful for displaying more subtlety in cases where there are many points and ranges are large.

3.3.2 Difference Maps

Difference maps are produced that visually represent a distribution’s difference from the background (or from any other distribution). These maps are derived by subtracting the likelihood of each bin in the background distribution from each bin in the candidate distribution. Results might therefore be negative, with positive numbers reflecting bins (or physical locations) where the candidate distribution is more likely than the background. A modified color spectrum is used in these difference maps, where zero is still depicted in black, but positive numbers utilize the warmer end of a rainbow spectrum (red, orange, yellow, and white) and negative numbers are depicted in cooler colors (blue, green).

3.4 Statistics

For a large taxonomy, it is useful to define some organizing principles in addition to the structure of the taxonomy itself. These principles can serve as a means of locating points of interest within the taxonomy - “handles” that one can grasp in order to pull out interesting features. To this end, various statistical measures were computed for each entry in the taxonomy.

The notation used is:

P = background, or the spatial distribution for all speech

Q = target word spatial distribution

n = number of observations

k = number of bins

i = bin index

Entropy

$$H(Q) = - \sum_i^k q(i) \log(q(i))$$

Entropy (or Shannon Entropy) is an information-theoretic measure that quantifies the amount of uncertainty in a random variable. In this context, entropy measures the degree of uncertainty about the location of an utterance, or how “spread out” a distribution is. For example, a distribution with all samples concentrated in a single bin would have 0 entropy, while a distribution with equal (non-zero) counts in all bins would have maximum entropy. Notice that entropy does not contain any information about spatial adjacency - a distribution with a single large peak (and otherwise uniform) would have similar entropy to one with many small peaks.

KL-divergence

$$KL(P, Q) = \sum_i^k p(i) \log \frac{p(i)}{q(i)}$$

KL-divergence, also known as relative entropy, measures how much information one distribution provides about another. In this context, it can be seen as a measure of the difference between two distributions. More specifically, KL-divergence is used here to measure how similar a particular spatial distribution is to the overall speech patterns in the home, or how unusual a particular distribution is.

Ripley's K

$$RK(\hat{Q}) = \lambda^{-1} n^{-1} \sum_i \sum_{j \in S_i} I(\hat{q}(j))$$

where:

$$\lambda = \frac{n}{k}$$

$$\hat{Q} = q(i) - p(i) \forall i$$

$$\hat{q}_i = q(i) - p(i)$$

$j \in S_i$ is the set of bins near bin i

and:

$$I(q(j)) = \begin{cases} 1 & : q(j) > T \\ 0 & : otherwise \end{cases}$$

This is a modification of the typical Ripley's K statistic [11, 7], originally designed to measure the degree to which a discrete spatial point process exhibits complete spatial randomness (CSR). Samples that are homogenous or those displaying CSR will have low values of Ripley's K, while those with tight clusters will exhibit high values.

Ripley's K was devised to measure the clusteredness of a set of discrete, unevenly spaced points by averaging the number of adjacent points in each cluster and normalizing by the overall density of the points. Here, I classify each bin as a point or not a point based on the residual probability after subtracting off the background. For each point i , evaluate $I(q(j))$ for each $q(j)$ in the neighborhood of i . $I(q(j))$ is an indicator function that is 1 when a bin has probability greater than T and 0 otherwise. T is a free parameter and is set to 0 in practice, but can be set differently in order to find different types of spatial clustering. When T is high, Ripley's K will give high scores only to distributions with clusters of high peaks. When $T = 0$ as here, the statistic has high value for distributions with clusters that are even slightly more likely than background.

Moran's I

$$I(\hat{Q}) = \frac{\sum_i \sum_j^n w_{i,j}}{\sum_i^n \sum_j^n w_{i,j}} \frac{\sum_i \sum_j^n w_{i,j} (\hat{q}(i) - \bar{q})(\hat{q}(j) - \bar{q})}{\sum_i \sum_j^n (\hat{q}(i) - \bar{q})^2}$$

where:

$$\hat{Q} = q(i) - p(i) \forall i$$

and:

$w_{i,j}$ is the weight between bins i and j . $w_{i,j}$ is a function of Euclidean distance between bins where bins that are further apart have lower weights. These weights can be thought of as the resolution at which the data is measured. In practice, $w_{i,j}$ is computed such that $w_{i,j} = 0$ when the distance between bin i and bin j is greater than 2 meters.

Moran's I [20] is a measure of spatial auto-correlation, or the correlation between probabilities in neighboring locations. The statistic is often used in fields such as epidemiology, where one would like to measure how much the presence of a point (i.e. a disease case) in one location affects the likelihood of a point in a nearby location. In this context, Moran's I measures the degree of smoothness in a distribution. The settings in the weight matrix ($w_{i,j}$) affect the scale at which smoothness is measured, where, for example, a distribution might be uneven at a fine scale, but display smoothness when more bins are considered simultaneously.

Moran's I values range from -1 (perfect dispersion) to 0 (random, no autocorrelation) to 1 (perfect correlation).

Entropy of Difference

$$H(\hat{Q}) = - \sum_i^r \hat{q}(i) \log \hat{q}(i)$$

where:

$$\hat{Q} = q(i) - p(i) \forall i$$

This measure is a test of how much entropy varies when compared to the background - distributions that are similar to the background will therefore display higher entropy in

their difference than will distributions with large variations from background.

Bhattacharyya Distance

$$D_B(P, Q) = -\ln(BC(P, Q))$$

where:

$$BC(P, Q) = \sum_i^r \sqrt{p(i)q(i)}$$

Bhattacharyya is a true distance metric (similar in some respects to Euclidean distance) that, similar to KL-divergence, is used here to measure a distribution's difference from background. Bhattacharyya distance is somewhat less sensitive to zero-count bins than KL-divergence, but provides a slightly weaker measure of difference in distributions with large n .

The effect of count

Many of the spatial distributions of interest contain too few samples to be robustly estimated, leading to poorly formed information theoretic measures. Furthermore, the measures that we can compute directly are extremely sensitive to the number of observed samples, making comparisons between distributions with varying number of samples difficult and often inaccurate.

For example, it can be shown rigorously that entropy decreases as a function of n - intuitively, the more samples you've seen, the more uniform a distribution will appear until, with large enough n it eventually converges to its "true" entropy. Likewise, with a single sample, the entropy of a distribution is 0, and this entropy increases with each subsequent sample until the distribution is adequately estimated and the true entropy is observed. These principles can also be modeled using artificial data; this empirical modeling was undertaken extensively as part of this work in order to understand the relationship between

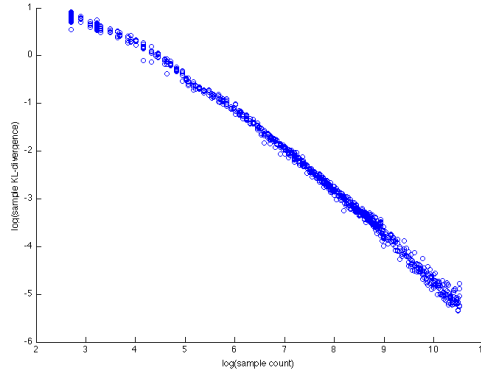


Figure 3-3: Sampled and observed KL-divergence

sample count and the various measures of interest. In all cases, measures were sensitive to n , and converged toward their true value as n increased.

First, the theoretical effect of n on $KL(P, Q)$ was derived [25]. This derivation established an upper bound on the expected KL-divergence of a distribution P against Q , which contains n samples. This expectation is:

$$\begin{aligned}
 E[KL(P, Q)|n] &= E[H(P, Q)|n] - E[H(Q)|n] \\
 &= H(P, Q) + \sum_i^B p_i \log(p_i - \frac{1-p_i}{n}) \\
 &\leq -\sum_i^B p_i \log(q_i) + \sum_i^B p_i \log(p_i + \frac{1-p_i}{n})
 \end{aligned}$$

Notice that $H(P, Q)$ is the cross-entropy of P and Q , which is unaffected by n . (x) implies that KL-divergence as a function of n converges toward the “true” KL-divergence with $\frac{1}{n}$; therefore the KL-divergence of a distribution against itself will converge to 0 linearly in log-log space, a property that can be verified by modeling. Figure 3-3 shows the observed KL-divergence of P with a distribution P_n generated by sampling P n times. Each P_n is generated m times, with all such KL-divergences plotted.

This relationship is also seen in actual data (see Figure 3-4).

Residuals

In order to overcome the effect of n on KL-divergence, the following method was devised (as part of related research [19]): the relationship between KL-divergence and n is linear in log-log space, therefore it is suitable to fit a line to these points plotted together and examine the residual from this line.

First, find $ax + b$ that minimizes sum of squared error of $\log(n)$ and $\log(KL(P, Q_i)) \forall i$.

Then for a given Q_i with sample count n , $KL_{predicted}(P, Q_i) = an + b$

And $KL_{residual}(P, Q_i) = KL(P, Q_i) - KL_{predicted}(P, Q_i)$

In simple terms, the residual effectively says “how does the observed KL-divergence for this word compare to the observed KL-divergence for words with similar counts?”

While KL-divergence and many of the measures discussed above are correlated with count, the residual measures computed here are uncorrelated with count, making it reasonable to use them to compare words with different sample counts. Figure 3-4 illustrates (L to R) the raw correlation of KL and count, the relationship between KL and count in log-log space with a line fit to the values, and the uncorrelated KL-residuals.

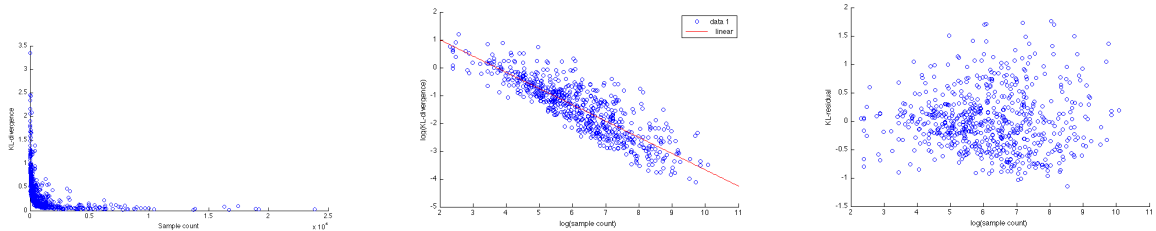


Figure 3-4: The relationship between count and observed KL-divergence

Notice that packed into this methodology are two possibly distinct effects - one is the effect of count on KL-divergence, which, as has been stated, can be rigorously proven. The second

is the semantic effect of word use on KL-divergence: it is possible that seldom used words are, in fact, used in ways that systematically differ more (or less) from the background, likewise with often used words. The residual measure can be thought of as a high level abstraction that embodies both of these properties in order to make a fair comparison between words.

Distribution Browser

Salient patterns can be seen in the visualizations described above even in very low resolution images, implying that interesting differences could be drawn out by looking at aggregate views of all distributions where each distribution is rendered at a small size. As a result of this observation, an approach was devised as follows. All spatial distributions are visualized as small, iconic heat maps and arranged according to some user defined ordering (i.e. alphabetical by target word). We then apply a statistical metric (i.e. KL-divergence) to each distribution, generating a score for each according to this metric. Icons are then darkened according to this score. The user can choose to visualize the scores in ascending (low scores are brighter) or descending (high scores are brighter) order. Additionally, the user can choose to filter the distributions by this score, showing, for example, only the top 50 scoring distributions.

Users can switch seamlessly between various statistical metrics, the ordering direction (ascending or descending), and the amount of filtering. The user can also choose to more closely examine any individual distribution in standard, log, or difference form. Additionally, an ordered list is provided for each metric that shows a total ordering of the target words based on the currently selected metric.

One can quickly get a sense of the shape of the distribution over the measure being examined. For measures that provide good separation between spatial distributions, the user

sees a uniform spread between dark and light icons. For a measure that clusters distributions toward one end of the scale, however, the user will see an even distribution in the dark (or light) part of the range, and just a few icons at the other end of the range.

As an example of the above effect, a particular measure gives a numerical score to “car” of .90. The next word, “diaper” scores .68. There are 15 words scoring between .02 and .50, and 408 words between 0 and .02. It is clear that most words have low scores, some have higher scores, and “car” is an outlier at the top of the scale. These properties are apparent when viewing the browser, as sorting in ascending orders shows nearly all icons as very bright, with just a few appearing dark, and “car” being black. Sorting in descending order is equally informative, as “car” appears very bright, several icons are less bright, and most icons are dark or black.

The browser allows the researcher to make informed decisions about the best statistical measure to use in order to select desired distributions. In the example of “car,” we were able to cycle through many measures quickly, noting in each case the position of “car” along the continuum from dark to light. We were similarly able to look for measures that highlighted words with similar spatial properties (in this case, words whose difference maps appeared tightly clustered in a particular location). As a result, we were able to conclude that the Ripley’s K statistic selects the desired spatial distributions. We could then use this measure to automatically sort the 658 target words, as well as any of the 26,000 other words in the corpus’ vocabulary.

Additional benefits are realized when we consider the ordering of the icons as a second dimension by which to view distributions, with darkening and lightening as the first dimension. Given the task of finding spatial information that is predictive of age of acquisition, we seek measures that are correlated with age of acquisition. In order to perform this search, we first order the distributions by age of acquisition, and then apply some measure. If correlation is high, we expect to see a smooth transition from dark (or light) at the top left to light (or dark) at the bottom right. Such a transition implies that measure values

are varying with age of acquisition. Figure 3-5 shows such an ordering for 120 words, with KL-divergence applied. We can see that KL-divergence values tend to be lower at the top left (distributions are darker) and higher at the bottom right. Although correlation is not perfect ($r = 0.58$), we can get a quick sense of the appropriateness of the measure. We can also quickly find outliers, or those distributions that are poorly predicted by looking for discontinuities in shading. For example, notice that “round” is far brighter than would be appropriate given its position in the matrix.

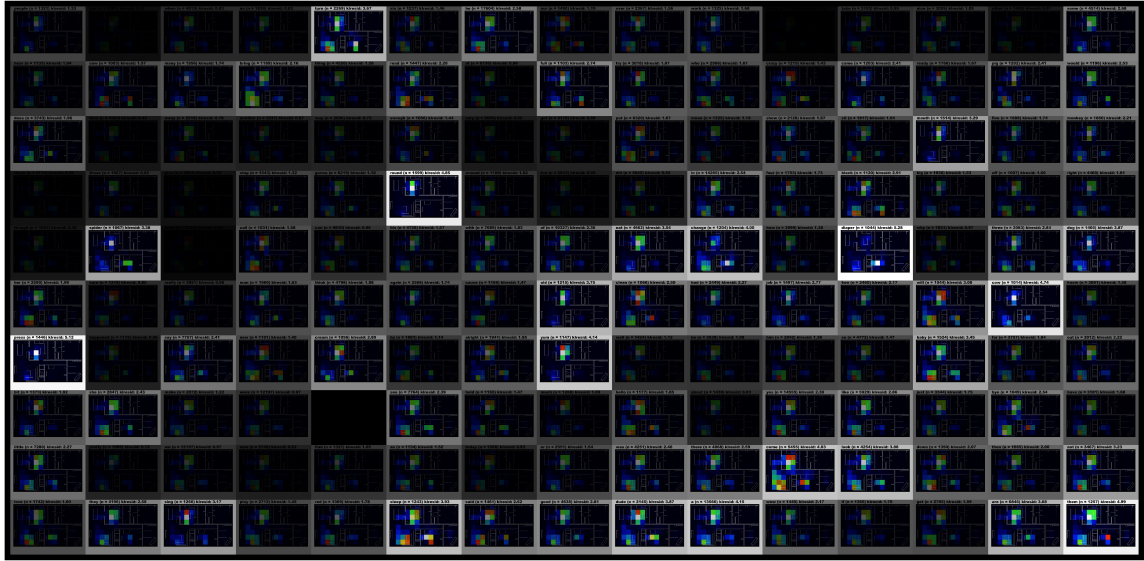


Figure 3-5: Difference Browser

Chapter 4

Exploration and Analysis

The taxonomy built up from spatial distributions is a useful tool for exploration and analysis, and in this chapter I will highlight some relevant pieces of data, showing that with careful comparisons, interesting insights as well as numerical results can be drawn out and analyzed.

4.1 Activity Types

Figure 4-1 shows heat maps representing 3 views of the overall activity pattern in the home. Even at a very rudimentary level, these visualizations provide insights about the daily life of the family. One can immediately see, for example, that the kitchen is a hub of activity, in particular the area near the center island. We can also see that a secondary hub exists in the living room near the couch, and that there are three main areas of the child’s bedroom where activity takes place, making up the third activity center in the home.

4.1.1 Speech vs. Movement

In Figure 4-2 we see the spatial distribution over all speech visualized in three ways. These heat maps show clearly several key areas of the home where speech is common (“social

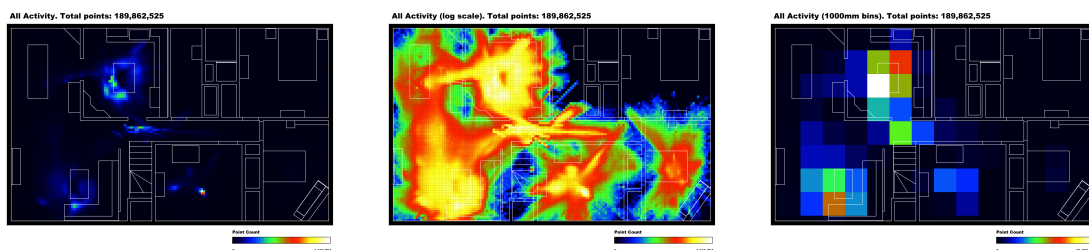


Figure 4-1: Heat maps: (L to R) all activity, all activity plotted on log scale, all activity with 1000mm bins

hotspots”): the kitchen, family room, and child’s bedroom. It is important to note that these hotspots were derived automatically via a very simple threshold-and-cluster algorithm that looks for high likelihood locations and builds clusters containing those locations, implying that these sorts of insights could be derived automatically.

Beyond knowing the locations of utterances, we might like to understand the ways in which speech acts differ in their locational properties from overall activity. In other words, are there locations in the home where people spend time silently? Are there locations in the home where people are seldom silent? These questions are answered easily by examining the difference map in Figure 4-3. We can see two prominent complementary areas in this map: in the kitchen near the left side of the center island, speech is likely relative to overall activity; and the hallway below the kitchen, where speech is unlikely. These observations make sense when we think about the activities that take place in these locations. In the kitchen as a whole, people may be moving around with little or no speech; however, during mealtimes (which take place at the left side of the center island) people are rarely silent. Likewise, in the hallway people are likely to be moving about silently as the hallway is not a place that one would tend to linger and talk.

4.2 Speech content

By examining speech on a per-word basis, we can begin to understand how particular words (and classes of words) fit into and are influenced by the patterns of daily life.

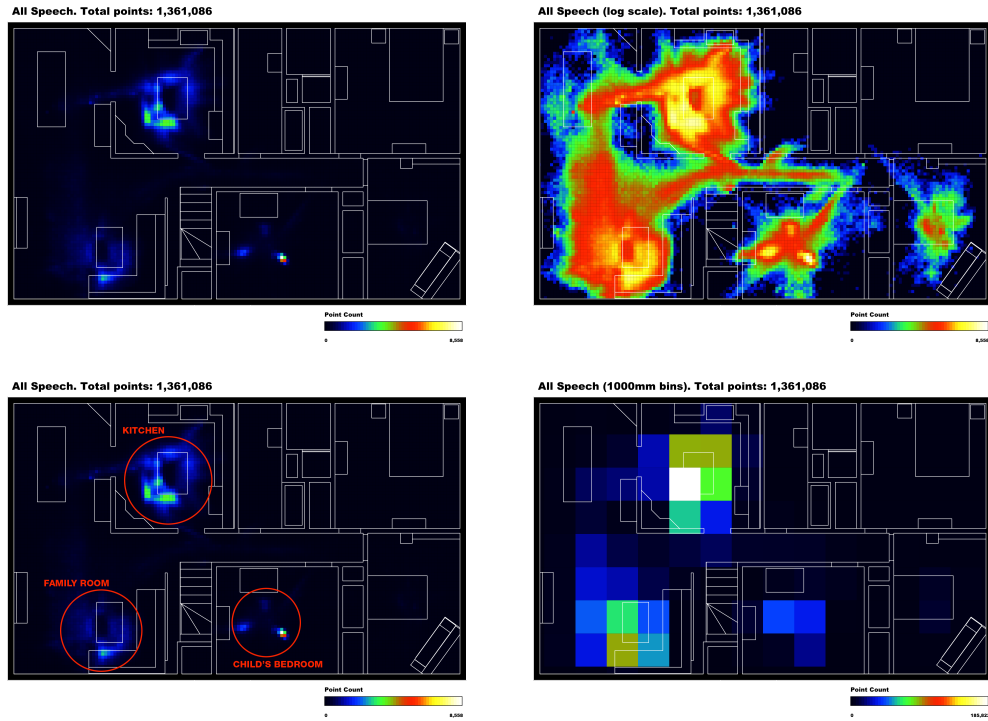


Figure 4-2: Heat maps: (Clockwise from top left) all speech, all speech plotted on log scale, all speech with 1000mm bins, social hotspots

4.2.1 Target Words vs. All Speech

Given this work’s interest in language acquisition, a natural focus is on the words that eventually entered the child’s vocabulary. Furthermore, we’d like to look at those words during the learning period (the time leading up to the child’s first production of the word) in order to understand if there are contextual cues that either facilitate or indicate the learning of the word.

Figure 4-4 summarizes the spatial properties of the 658 target words, as used during the learning period for each word. The key insight from these visualizations is the existence and location of two “learning zones,” or areas where the child was taught much of the language he came to know by the age of two. These are the areas where these words were used most often, making it reasonable to assume that the learning process took place in these areas

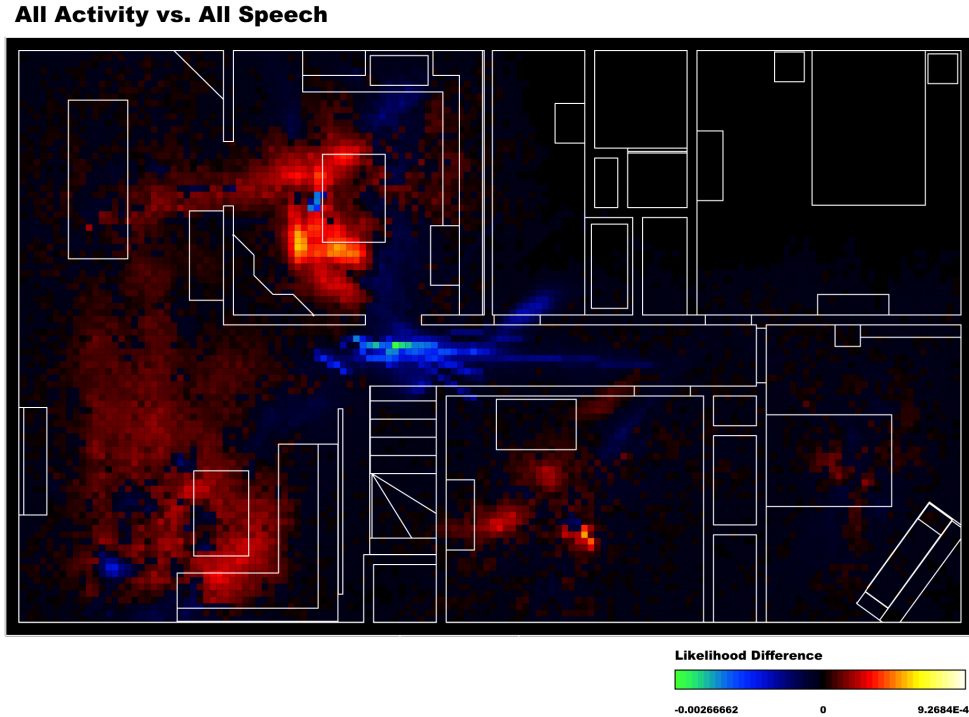


Figure 4-3: Difference map showing speech vs. all activity

primarily.

4.2.2 Spatial groundedness

A word that became a focus because of related research was “car.” This word reduced the perplexity of a spatial language model more than any other word, implying that spatial properties of the word were important. The spatial distribution for car appears to follow a typical usage pattern, with the word showing up in many areas of the home. This usage pattern differs significantly from the overall speech pattern in the home, however; a difference that shows up immediately in the difference map visualization - the area near a window in the family room appears bright yellow and orange, with the rest of the house being blue, black, and green.

This pattern shows that “car” is used normally or less than most words throughout the

home, but is far more likely than other words in the area near the window in the family room. This pattern is intuitive for a researcher familiar with the data: the child often stood at the window with his nanny, pointing to cars as they drove by. There was also a play mat near the window where the child often played with toy cars. A word whose usage

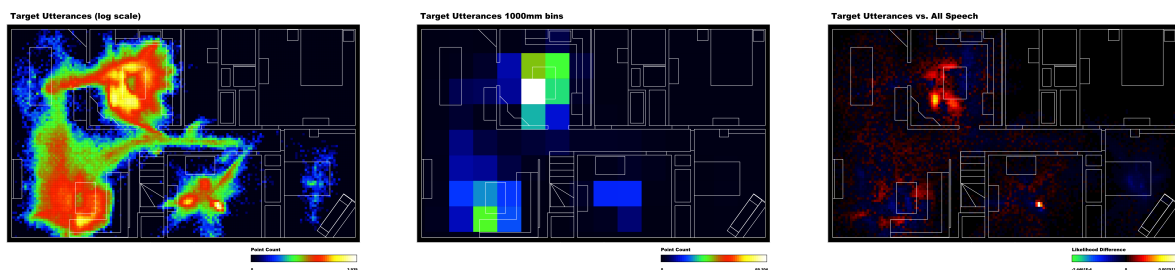


Figure 4-4: Target word heat maps

pattern is similarly localized is “diaper.” This word, as might be expected, occurs far more frequently near the child’s changing table than in other parts of the home. This pattern is again evident upon examination of the difference map for the word.

Several food-related words also follow similar patterns, again, as expected. Words such as “mango,” “banana,” and “papaya” occur far more frequently near the child’s primary feeding location in the kitchen. Similarly, but as a slight variation, several words including “eat” and “done” (part of the phrase “all done”) occur throughout the kitchen, but with a more varied spatial distribution than words that tend to occur strictly while the child is eating.

By contrast, there are many words that are spread more uniformly throughout the house. Words such as “you,” “those,” and “that” exhibit spatial distributions that mirror closely the distribution of all speech. These words and many like them are not tied to particular locations, which is an intuitive property when one considers the meaning of the words. Words that fall into this class are generally words that describe moveable objects, people, or concepts, none of which are tied to locations. Of equal interest are words such as “come” and “go,” whose spatial distributions are spread throughout the home with the exception

that they occur infrequently in the kitchen where the child was often confined to a high chair and thus was unable to “come” or “go.”

The Ripley’s K statistic is intended to measure the “clumpiness” of a set of points, or the degree to which a set of points exhibits complete spatial randomness (CSR). Here, Ripley’s K is applied to difference distributions using a threshold on the probability to determine which bins are considered “points” (see Figure 3.4). Those distributions that are more likely than background in localized ways therefore have high Ripley’s K values.

Ripley’s K can be an effective handle into the data - by searching for words with similar patterns of clustering, we can find those words with similar ties to locations in the home. To find words whose usage is grounded in a particular location, for example, we need only find those words with high Ripley’s K values (see Figure 4-5).

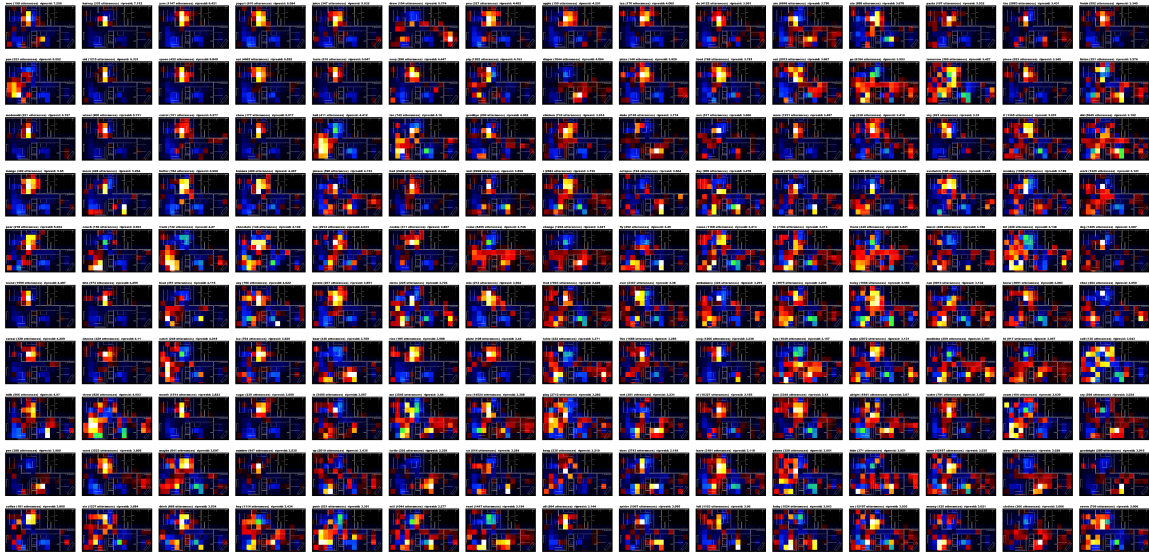


Figure 4-5: Top 150 Words by Ripley’s K

This is a powerful concept. By pulling out those words that are tied to locations while being able to recover those locations, we have the opportunity to begin to derive meaning for a certain class of words simply by looking at the usage patterns for those words. Because these distributions are aggregated over long periods of time (ranging from weeks to years),

we can assume that these spatially tied words relate to either objects or concepts that are locationally invariant to some extent. Diapers are always present in the area of the child’s room where “diaper” occurs most frequently, mangos are always cut in the same area of the kitchen, etc. It is therefore possible in principle to recover via visual information a description of the items being discussed.

To summarize, we can take all speech in the home and, via spatial distributions alone, highlight those words that are tied closely to particular locations. We then might search these locations visually for the object or concept that the word describes, providing true grounding for the word in an automatic way. Figure 4-6 illustrates this concept using video frames taken during utterances containing the word “ball.”

A problem with this approach arises when we consider words that are used in specific locations exclusively, but that relate to objects or concepts that are not visible at that location. An example from this data is the word “bus” which was used often in the kitchen and has a high Ripley’s K score, but that was part of a mealtime song about a bus. There is no visual clue to be found that relates to “bus.”



Figure 4-6: Snapshots taken during utterances containing “ball” in the location associated with “ball”

4.2.3 Clustering

Clustering is an effective tool for exploring the relationship between Ripley’s K and KL-divergence, and how these measures might relate to the meaning and natural usage of words. Words with high Ripley’s K and also high KL-divergence, for example, would be those words that are focused in locations that are substantially different from overall speech. Similarly,

words with high Ripley’s K and low KL-divergence would be words whose locational focus is well represented in the overall speech pattern. An example of the former is the word “diaper” whose usage is highly focused in a location that is not a center of overall speech. The word “mango” is an example of the latter - “mango” is used often in a single location, but that location is well represented in the background (the kitchen near the center island is the single most active speech location in the home); “mango” therefore has a relatively low KL-divergence. A final example is “them ” which has a high KL-divergence and a low Ripley’s K score, implying that this word is used in a way that stands out from background, but is not tied to any single location.

In order to explore the ability of multiple spatial features to segment words into salient groupings, K-means was applied to the data in two dimensions where KL-divergence is on one axis, and Ripley’s K is on the other. K was set to 30. See Figure 4-7 for visualizations of the clusters generated. Some interesting examples of the various clusters follows.

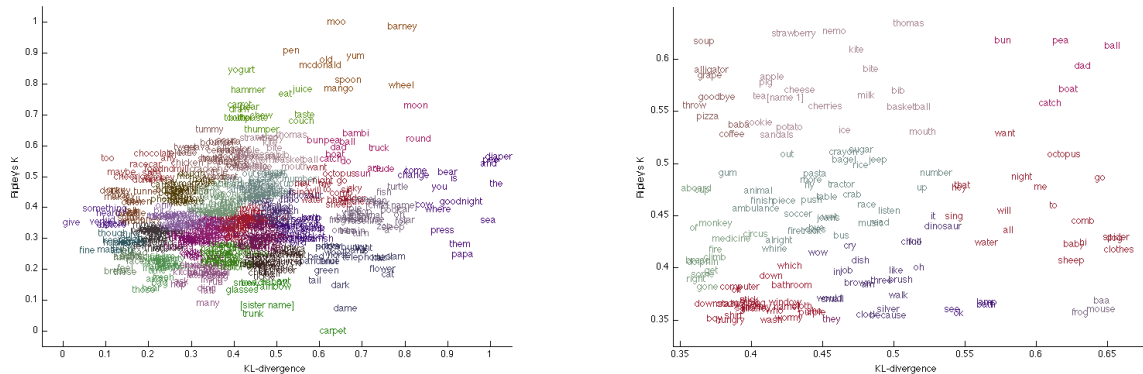


Figure 4-7: K-means clustering by KL-divergence and Ripley’s K

The cluster with centroid (.64, .88), which represents a relatively high KL-divergence and the highest Ripley’s K of all clusters contains the following words:

- “mango”
- “spoon”
- “yum”
- “old”

“mcdonald”

This cluster of unusual, spatially tied words appear all to relate to mealtime (the song “Old McDonald had a farm” was a mealtime favorite). This cluster provides evidence that meals are the single most unifying factor in language use - no other activity in the home exhibits such strongly spatially tied words, or as many words that differ so significantly from the background. The high degree to which these words are spatially tied relative to words in other locations or related to other activities might be explained by the fact that during meals, participants are generally seated. In particular, mealtimes are one of the only times that the child is stationary for extended periods.

As an example of the effect of mealtime, we examine another cluster, this one with centroid at (.90, .44). This cluster diverges more from background than the previous one, but is less spatially focused. Words in this cluster include:

“come”

“goodnight”

“change”

“diaper”

“where”

“you”

These words vary in the ways that they are used, both spatially and in the activity contexts they are part of. All exhibit moderate spatial clustering, which is clear for “diaper,” “goodnight,” and “change” but is somewhat less obvious for “come,” “where,” and “you.” Visual examination of these latter three words’ distributions reveals that the usage of these words is, in fact, clustered, but not in a single location. “come” has a cluster in the child’s bedroom and another in the living room, while “you” shows a cluster in the child’s bedroom, another in the kitchen, and a third in the dining room.

A final example comes from the cluster with centroid (.10, .28). This cluster should contain words that resemble background and that are not location-specific, and indeed it does:

“about”

“keep”

“fine”

“sure”

“need”

These are words that are more grammatical in nature, which would be words that would be expected to be used in a variety of contexts.

These observations provide some insight into both the statistics and the usage of these words. KL-divergence is capable of measuring various disparate properties of a word’s distribution - words that are unusual may be unusual in various ways. Ripley’s K, on the other hand, appears to be measuring the single property that it is intended to measure - the degree to which a word is tied to a single location. High scores are typically found with words that are tied to one location, while moderate scores appear tied to several locations, and low scores are spread more uniformly.

What these statistics reveal about the usage of the words is slightly more difficult to quantify. We can see that words without spatial ties and low KL-divergence tend to be more general words, and the words with moderate KL and high Ripley’s K tend to be highly focused, specific words. But the words in the middle group with high KL and moderate Ripley’s K are more individually different. “come” has a different reason for displaying the values it displays than does “goodnight” or “diaper.” Each word essentially has its own story.

4.3 Identity

It has been shown that peoples’ identities can be accurately segmented into classes using a combination of behavioral traces (data from person tracking) and visual features (color histograms from video) [31]. In this work I focus on a much coarser representation, spatial distributions, but propose that they still contain enough individually identifiable information to be useful for identification.

As just one example of a person-specific feature, consider the area around the kitchen island. Each caregiver has a location that they prefer, a fact that can be verified by watching video of mealtimes. The mother tends to sit close to the bottom edge of the island, while the father prefers the left side, and the nanny, who is often alone with the child at mealtime, sits nearer the corner of the island. When we examine the difference maps in Figure 4-8, these preferences are apparent - the mother is far more likely to speak in her preferred location, the father in his, etc.

I currently make no claims as to a quantitative assessment of this concept, however it appears reasonable that we could derive an aggregate distribution for each person of interest and then generate at least a prior if not a full classification of identity based on a small sample of observed data. In keeping with the cross-modal intent of this work, this prior could be used in conjunction with an audio-based speaker ID system to improve classification. This effectively says “where something was said influences who I think said it.”

4.4 Temporal slices

An interesting feature to notice in Figure 4-9 are the various activities that can be seen clearly in this simple comparison. The morning shows the mother feeding the child (the hot spot is associated with the mother’s usual feeding location, see Section 4.3 above). Daytime shows the nanny spending time in the chair in the child’s room and near the window, as well as meals in the nanny’s usual location. Evening shows meal preparation, which differs from breakfast and lunch in that it is spread throughout the kitchen. Presumably this is because there are often two adults preparing the meal, moving around the kitchen cooking and so on, and because preparation of the meal is more involved than with breakfast or lunch. At night there is nearly no activity in the kitchen, because the family is spending much more time on the couch in the living room.

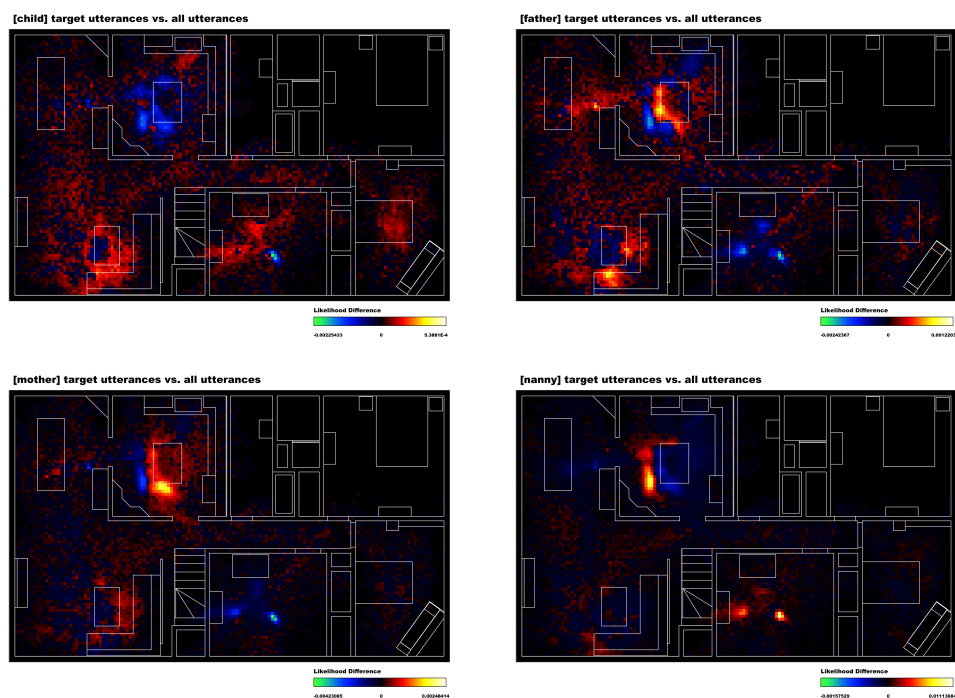


Figure 4-8: Difference maps for (clockwise from top left): child, father, nanny, mother

4.5 Age of Acquisition correlation

A key question that we might ask of this dataset is whether there is information contained in spatial distributions that indicates the acquisition of language in the child. More concretely, are statistical measures of spatial distributions for individual words correlated with the age of acquisition for those words? If this correlation does exist, then we can say at least that there is some relationship between where words are said and when the child learns them. This relationship is likely to be complex, as we are dealing with a dynamic system involving several people who are constantly influencing each other in multiple feedback loops. A straightforward causal relationship is unlikely in such a “loopy” system, but correlation would be informative nonetheless.

Previous work has looked at the effect of the frequency of word use on age of acquisition. Previous work on HSP has verified that this relationship with frequency exists

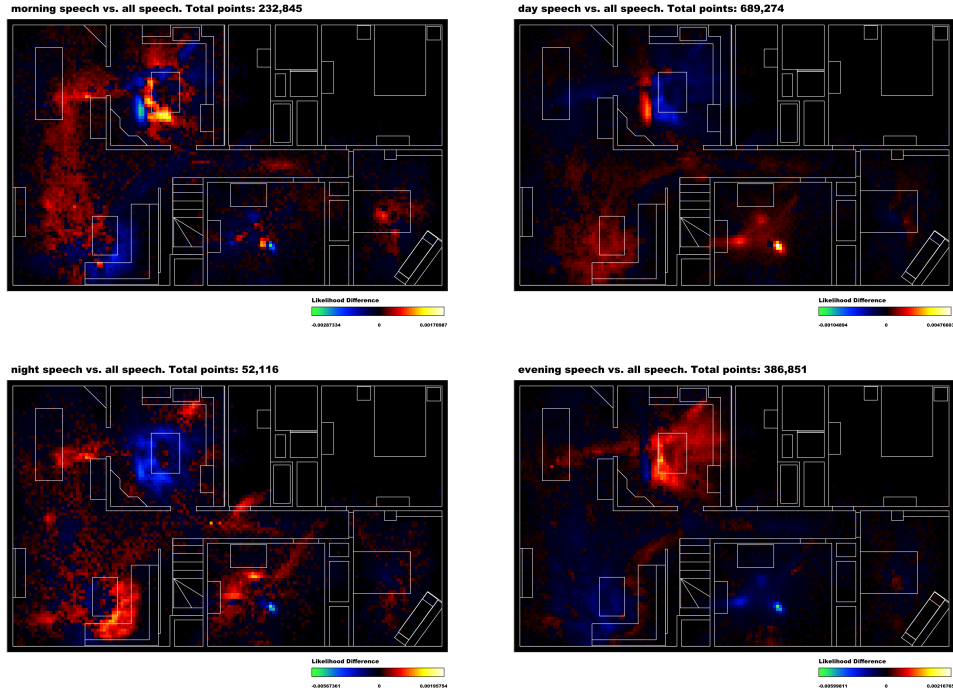


Figure 4-9: Difference maps for (clockwise from top left): morning, daytime, evening, night

in this dataset as well, and has added similar correlations with prosodic features and AoA [37]. This work builds on those concepts, looking for correlation with spatial data.

The basic prediction methodology is as follows:

1. Take the background spatial distribution representing all adult speech. Call this P .
2. Take spatial distributions for each target word's learning period. Call these Q_i .
3. Compute some measure M_j (i.e. KL-divergence) for each $Q_i : M_j(P, Q_i)$
4. Using a least-squares linear regression, fit a line to each $M_{i,j} \forall i$ plotted against AoA_i
5. Pearson's r values are reported as r_j

Several of the measures applied to spatial distributions are predictive of AoA. The highest correlation for all 658 words is KL-divergence (note that this is actually KL-residual, described previously), with $r = -0.41$. We can see that with even a small amount of filtering, Ripley's K dominates the other metrics in terms of prediction accuracy. Ripley's K

(initial $r = -0.33$) reaches an early peak when words with less than 1825 samples are excluded and $r = -0.81$. At this level of filtering, we predict only 96 of the original 658 words.

Figure 4-10 shows each r_j as a function of a sample count (n) threshold T : words for which $n < T$ are discarded for $0 < T < 8750$.

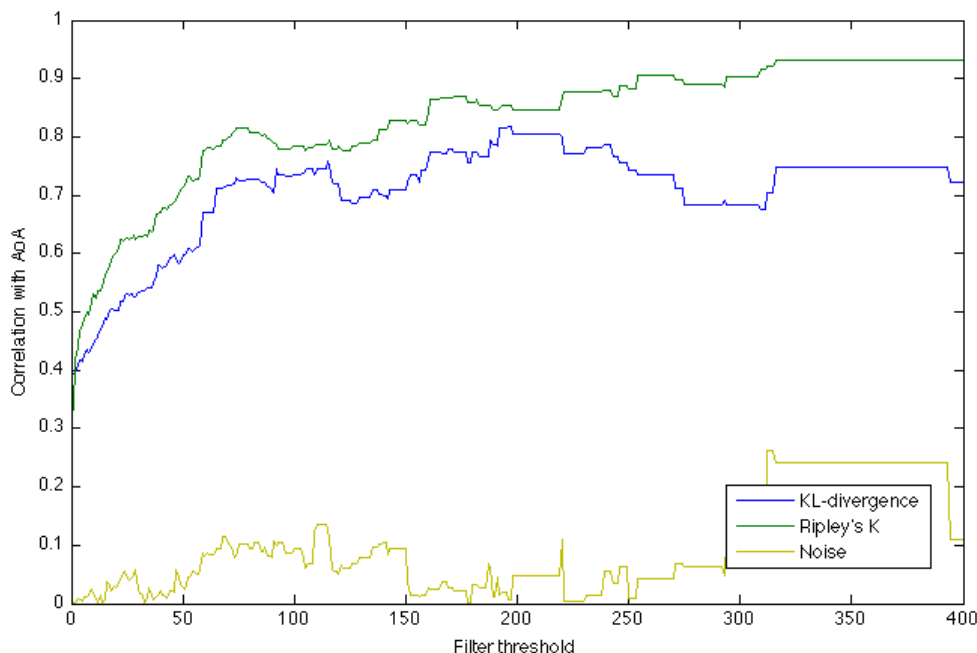


Figure 4-10: Predictor accuracy as a function of sample count threshold

As this is an early result, it is still unclear why prediction goes up as much as it does when we filter by count. It is possible that the high count words are simply better estimated than lower count words, and so are more accurately predicted. Or, it is possible that higher count words are more sensitive to spatial usage patterns. It is also possible, however, that filtering is introducing a subtle confound to the regression model. This is an interesting area for further research. For the remainder of this section, however, I focus on prediction over the full set of 658 words with no filtering.

If we take an existing known predictor, frequency ($r = -0.35$) and construct a regression model with frequency and KL-residual ($r = -0.41$) the correlation coefficient of this

multivariate model is $r = 0.50$ ($r^2 = 0.25$), showing that there is information in the spatial distributions that is not contained in frequency, and that these two predictors together can achieve a high correlation with AoA. Figure 4-11 shows the full prediction of this model (linear fit of prediction vs. actual shown in red, diagonal shown in grey).

It is worth investigating the correlation with spatial features further, so I now again remove frequency from the model in order to assess KL-residual on its own. The full prediction for all 658 words is shown in Figure 4-11, as well as the best predicted half of the words ($r = 0.89$) and the worst predicted half ($r = 0.11$).

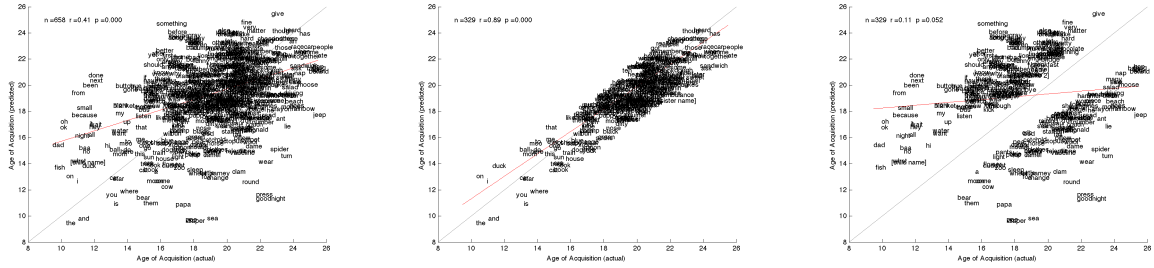


Figure 4-11: KL-residual correlation with AoA

Words with usage patterns that differ more from the overall language patterns in the home tend to be learned earlier by the child, and as Figure 4-11 shows, some words are much more sensitive to this effect than others. Are these usage patterns driving the learning of the word by the child? Or are they reflective of the process of word learning in the child, a process that is driven by some other force? Language learning is a complex process, and this is a difficult question to answer quantitatively, so one can only speculate and attempt to provide evidence.

I will argue that a mix between the two is true - the learning of any particular word by the child is driven primarily by practical goals and desires and what we see in the spatial distributions is reflective of the caregivers' use of the word in a child-directed way; and to a lesser extent words that are spatially unusual are more readily learned by the child, perhaps due to an effect like Bruner's formats [2].

The child has a need to communicate in order to get food, toys, and to socialize with his caretakers, and these are some of the forces that drive his learning. His inherent interests are what cause him to learn words like “car” and “truck” earlier, while his desire to be fed causes him to learn “mango” and “cookie.” This again is a system of loopy causality, where the child’s goals are reflected in the actions of his caregivers, and the goals of the caregivers are reflected in the actions of the child. We can simplify this system, though, and say that a reason exists to learn a particular word, and because of the dynamic nature of the interaction between caregiver and child, this reason is reflected in the way the word is used, which manifests as a statistical difference in the spatial patterns around the use of the word.

Another way to think about this potential explanation is that a word might be used in one of two ways - either in an “adult” way, or in a “child-centric” way. It is then reasonable to think that the degree to which a word is used in a child-centric way would be correlated with the age at which the child learns the word - words that are often directed at him would be expected to be integrated into his vocabulary earlier. This argument rests on the assumption that the use of a word in relation to the child is different (and furthermore is different in a way that can be quantified using the methodologies described in this document) from the way an otherwise similar word would be used between adults. If that were not the case, then the spatial distribution of a word that the child learned would not differ from that of a word the child did not learn.

As a crude test of this hypothesis, we can first make the assumption that the best estimate of adult speech patterns comes from the child’s parents. The nanny spends significantly more time alone with the child than either parent, and so uses language in a more child-directed way. Visitors to the home are likely to use language in a way that both differs from normal speech patterns and that is more likely to be directed at the child (when Grandma comes over, for example, she is likely to spend significant time addressing the child). And, of course, the child’s speech is a poor estimate of adult speech.

We can therefore construct a background distribution containing only the parents’ speech as

a proxy for adult speech. If the correlation with KL-divergence is in fact measuring at least in part the amount to which a word’s usage patterns are “child-centric,” we would expect that effect to be amplified when KL-divergence is measured against this somewhat purer adult speech background. And this is, in fact, what I found. When KL-divergence is computed against the adult background (as opposed to the background representing all speech, as was previously described), we see a correlation of $r = -0.45$ as opposed to $r = -0.41$ with the standard background. This is surely a crude test, but does provide a small amount of evidence to support the notion that KL-divergence is encoding the “child-centric” use of a particular word.

We can also probe this effect from the other direction. Take only the nanny’s utterances for a given word and compare that distribution to the background, again assuming that the nanny’s language use more closely resembles child-centric speech than any other’s. If the nanny’s speech is uniformly child-centric, then we would expect this comparison to contain only the differences due to the latter effect described earlier - that is, the spatial distributions reflect only the degree to which a word’s usage is unusual as a function of its meaning, not the degree to which it is child-centric. If my original hypothesis holds, then this correlation should be lower, and indeed it is with $r = -0.22$. Because this comparison presumably does not contain variation due to child-centric use of words (it is all equally child-centric) we would also expect a lower variance in the KL-divergences, which we also see ($\sigma = 0.42$ vs. $\sigma = 0.58$ for the original KL-divergences). As before, this test provides a small amount of evidence to support child-centricity as the primary piece of information contained in KL-divergence, with spatial difference also correlated with AoA, but to a lesser extent.

It is important to attempt to understand the forces guiding the child’s learning of words beyond what is reflected in the spatial distributions, and a way to do this is to first look at words that are predicted poorly by the model. First I’ll define the error metric by which I measure how well the model predicts a word. Because words in the center of the range are more likely to have lower prediction error (there is simply less room for a mistake), I nor-

malize error by the maximum possible error, given a word’s true AoA. Error for predicted age of acquisition AoA_p in relation to actual age of acquisition AoA_a is therefore:

$$\epsilon = \frac{abs(AoA_p - AoA_a)}{max(AoA_a - min(AoA_a), max(AoA) - AoA_a)}$$

If we look at the two words that are predicted most poorly by KL-divergence, “pee” and “diaper,” we can get some idea about these forces. These words are highly localized in their usage and have high KL values and so are predicted to be learned early by the child. These words are presumably uninteresting to the child, however, and are unlikely to be encouraged by the caregivers and as a result were learned much later than predicted. Similarly, “maybe” is predicted by this model to be learned late (it is used in a way that resembles all speech) but it is in fact learned earlier. This is possibly because the word is useful to the child, garnering his interest. Likewise, “dad” is predicted by the model to be learned much later than it was actually learned, presumably because this word is quite important to the child (as with many children, “dad” was the first word learned by this child). These cases all provide evidence that there is some other force (i.e. interest) guiding the child’s acquisition of words, and that the spatial distributions reflect the ways in which words are used around the child, but are wrong in cases where the child’s interest level (either high or low) is incongruous with how the word is used.

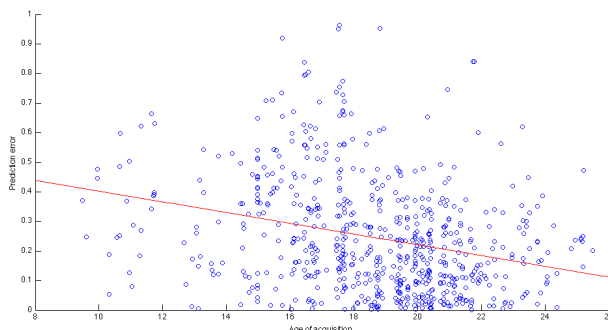


Figure 4-12: Prediction error vs. actual age of acquisition

Figure 4-12 shows that on average, words that were learned earlier are more poorly predicted by the model. This implies that there is some other motive for learning these words that is transparent to this model - there is no evidence from the way the word is used that

it should be learned as early as it is actually learned yet the child's interest acts as a force for word learning.

One might further argue that words that are learned later are less subject to the child's interest as a force for learning, since the child's vocabulary is broader and communication is easier for him as he gets older - he has less of an intense need to learn new words, therefore other forces drive his learning more. These other forces would include spatial usage patterns (whether due to semantic needs, child-centric usage, or other effects), implying that spatial statistics would better predict words that were learned later, which is in fact what we observe.

As a final window into these forces, it might be useful to examine words that the child did not learn by 24 months. The words "microwave," "appointment," and "quarter" are all words that appear to be uninteresting to a child. They all have relatively high KL values however (1.02, 1.10, and 1.13, respectively), and would be predicted by the model to be learned at approximately 16 months in all cases. Because of the lack of appeal to the child, however, none were learned before 24 months.

We have seen that there is some force that is influencing the child's learning of various words beyond what can be seen in spatial or linguistic properties. This force is presumably practical - regardless of where, how, or how often a word is used, the child's desire to learn that word exists on an independent graduated scale. These other factors (frequency, spatial properties, etc) likely have some influence, but these other forces must be taken into consideration when attempting to understand language acquisition. It also appears likely that KL-divergence, or the degree to which a distribution differs from overall speech patterns, contains information about how a word is used in relation to the child. There is possibly some effect of these spatial properties influencing learning, but it is likely that a large part of the correlation we see is not causal, but a secondary effect.

Chapter 5

Conclusions

5.1 Contributions of This Work

This work represents the first ever large scale, comprehensive look at movement patterns and language use in daily life in a natural setting. In it, I showed how to construct a large multi-modal dataset from raw video and audio, developing scalable algorithms for various aspects of processing. Most notably, I developed a system to perform accurate, efficient person tracking, and data structures for aggregating, visualizing and analyzing tracker output in relation to other modalities.

This work showed that spatial properties of language use conveys information about the participants, the activities in which language is embedded, and in some cases the meanings of the words. With a suitable roadmap based on visualization and descriptive statistics, one can test hypotheses, formulate new questions, and derive meaningful insights and numerical results from this dataset. It was shown that not only are the spatial properties of language use relevant in the ways we might expect, but that more subtle information is lurking just beneath the surface as well.

5.2 Future Directions

There are many sources of potential error in the methods described here. Most notably, tracking people in video is a difficult problem and the person tracks produced by 2C are imperfect. While algorithms exist that can produce more accurate tracks, these algorithms are too computationally expensive to be applicable to this corpus. As machine vision progresses and hardware speeds increase, however, we can expect the bar to be raised in terms of what is possible at scale.

The added precision of more accurate tracking might improve the results described here, but could also open up new research directions that are currently impossible - following subjects for long periods, for example, could lead to new insights into sequences of behavior and longer causal chains in regards to language use.

Another important source of error in this work comes from speaker identification. If speaker ID were perfect, for example, age of acquisition would not be a source of potential error - rather than implementing an algorithm to derive age of acquisition, we could just query the database. A worthwhile goal to pursue would be deriving accurate identification from video data (perhaps in a multi-modal system that integrates information derived from audio as well). With accurate person identification based on both audio and video, a researcher would have the ability to study in detail and at large scale the interaction patterns between people both in relation to language and not, again with the ability to understand long causal chains and complex dependencies.

Many of the insights discussed in the Exploration and Analysis chapter would be fertile ground for further research. For example, the simple clustering scheme I described is only a very coarse view of the way in which words relate to each other spatially. More sophisticated methods were explored, but not developed fully and it isn't difficult to imagine that a more comprehensive approach might be developed that groups words in even more interesting, salient ways.

This thesis leaves many compelling questions unanswered. For example, how does the child's language use change over time? Can we see how his comprehension increases after a word is learned from the spatial properties of his use of that word? How do the movement patterns of one individual relate to those of any other individual, and do those relationships provide insights into language use?

A strong consideration in many of the design choices I've made was that the dataset and methodology be general enough to be usable by others in relation to the research directions described above as well as in pursuit of goals that I've not thought of. My time with the Human Speechome Project has ended, but it is my hope that this work provides a firm foothold for future researchers working on the project.

Appendix A

Data for Target Words

This appendix gives quantitative data for each of the 658 words that were in the child's vocabulary by the age of two. All data is for the learning period of the word - that is, the period before the child's first production of the word. Data given is age of acquisition in months, the number of utterances containing this word, the number of location points in the spatial distribution for the word, the KL-divergence (normalized) of the spatial distribution, and the Ripley's K value (normalized) of the spatial distribution.

Table A.1: Data for words in the child’s vocabulary

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
a	15.624	13,566	16,295	0.752	0.326
aboard	17.646	69	105	0.367	0.476
about	16.458	1,439	1,929	0.168	0.261
accident	20.410	76	108	0.330	0.499
after	15.456	293	368	0.173	0.326
again	20.313	2,566	3,216	0.316	0.377
air	21.710	192	239	0.323	0.256
airplane	17.548	170	234	0.303	0.239
album	20.814	11	12	0.273	0.307
[name 1]	19.956	6	10	0.368	0.563
[nanny name]	15.456	407	495	0.326	0.363
all	11.642	994	1,183	0.550	0.436
alligator	18.242	36	42	0.333	0.590
alright	19.380	1,841	2,461	0.353	0.426
am	20.342	698	846	0.466	0.384
ambulance	21.523	324	371	0.394	0.457
an	23.755	1,830	2,331	0.150	0.351
and	11.025	2,998	3,369	0.974	0.541
animal	19.543	373	428	0.405	0.474
another	22.710	1,237	1,582	0.122	0.241
ant	22.978	34	36	0.521	0.159
any	16.056	796	1,014	0.302	0.558
anything	16.056	462	606	0.244	0.366
apple	15.313	155	178	0.437	0.583
are	14.986	6,846	7,960	0.667	0.525
around	20.718	1,108	1,522	0.184	0.284
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
as	18.524	1,134	1,511	0.276	0.292
ask	23.876	715	815	0.235	0.159
at	21.708	6,736	8,848	0.162	0.345
ate	25.489	888	1,124	0.276	0.510
away	22.942	2,218	2,819	0.142	0.167
awesome	21.226	61	78	0.236	0.300
baa	11.083	176	190	0.672	0.369
baba	14.977	24	31	0.376	0.537
baby	15.756	1,524	1,742	0.625	0.422
back	14.453	887	1,115	0.306	0.329
bad	17.557	598	819	0.174	0.252
bag	17.695	161	227	0.203	0.154
bagel	21.344	48	92	0.483	0.504
ball	12.925	411	512	0.649	0.612
balloon	17.714	287	341	0.473	0.328
bambi	18.579	51	64	0.632	0.640
banana	20.313	490	712	0.435	0.612
barney	18.944	335	379	0.804	0.987
basket	20.215	165	208	0.542	0.306
basketball	20.890	94	139	0.496	0.554
bath	16.562	375	471	0.502	0.365
bathroom	18.754	124	183	0.349	0.383
be	16.864	3,538	4,868	0.195	0.370
beach	23.512	133	131	0.396	0.155
bear	14.555	430	440	0.850	0.515
beautiful	21.211	567	676	0.303	0.385
because	10.645	88	111	0.443	0.354
bed	18.514	454	569	0.513	0.241
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
bee	17.578	181	191	0.413	0.211
been	11.325	186	230	0.313	0.302
beep	22.388	236	319	0.256	0.447
before	16.449	501	687	0.156	0.343
beginning	20.147	92	118	0.175	0.323
behind	24.945	279	330	0.236	0.287
being	19.923	467	607	0.145	0.292
bell	16.883	80	83	0.447	0.185
better	15.727	391	509	0.098	0.297
bib	19.913	91	128	0.523	0.569
bicycle	18.514	272	315	0.519	0.220
big	17.549	1,936	2,554	0.277	0.342
bird	16.717	779	858	0.549	0.285
bit	21.140	1,373	1,963	0.348	0.388
bite	20.813	574	766	0.559	0.590
black	17.953	1,120	1,365	0.528	0.331
blanket	13.159	16	18	0.400	0.290
blue	16.043	463	497	0.565	0.226
boat	16.847	351	438	0.622	0.571
body	20.980	143	165	0.301	0.303
boo	15.490	182	219	0.354	0.328
booger	17.695	73	100	0.298	0.253
book	14.978	716	807	0.710	0.390
boom	16.153	166	209	0.308	0.327
bottle	19.643	374	524	0.171	0.380
bounce	19.479	70	99	0.337	0.608
bowl	22.957	255	309	0.339	0.384
box	19.449	297	367	0.439	0.293
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
boy	14.818	914	1,121	0.321	0.351
bread	14.986	106	149	0.397	0.407
break	19.445	306	393	0.091	0.192
breakfast	18.977	176	283	0.288	0.346
bridge	19.612	38	50	0.224	0.455
bring	23.187	1,189	1,642	0.391	0.316
broke	20.409	249	334	0.186	0.370
brother	19.693	106	139	0.276	0.218
brown	16.747	256	269	0.427	0.386
brush	16.447	140	163	0.455	0.388
bubble	15.189	48	59	0.317	0.277
buddy	21.224	136	192	0.165	0.208
bug	17.048	145	183	0.268	0.324
bum	17.552	64	78	0.200	0.240
bump	16.755	139	168	0.575	0.334
bun	16.594	60	71	0.571	0.618
bunny	18.580	220	241	0.606	0.271
burp	24.828	247	320	0.188	0.209
bus	14.687	76	89	0.442	0.431
but	14.515	1,785	2,230	0.400	0.334
butter	23.311	184	244	0.446	0.689
butterfly	18.747	256	261	0.460	0.302
button	13.290	97	101	0.340	0.390
by	19.923	1,061	1,398	0.206	0.365
bye	15.024	1,049	1,190	0.460	0.438
cake	20.815	269	336	0.351	0.362
call	21.358	1,031	1,357	0.286	0.339
came	19.579	1,203	1,368	0.436	0.222
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
camel	18.579	62	63	0.662	0.426
camera	16.152	102	134	0.505	0.266
can	20.916	9,810	13,717	0.175	0.435
car	12.918	479	540	0.763	0.394
careful	20.244	687	895	0.262	0.249
carpet	20.858	11	13	0.574	0
carrot	18.747	101	127	0.416	0.732
cat	14.708	667	684	0.755	0.183
catch	18.513	248	347	0.595	0.558
cause	19.945	1,168	1,676	0.266	0.474
cell	21.942	136	194	0.155	0.422
cereal	19.419	328	437	0.380	0.584
chair	14.978	213	277	0.359	0.219
change	18.790	1,204	1,629	0.724	0.503
chase	19.693	65	91	0.309	0.184
check	20.275	382	535	0.257	0.247
cheerios	21.843	24	11	0.104	0.498
cheese	19.481	429	597	0.499	0.570
cherries	21.654	54	72	0.450	0.555
chew	17.727	177	219	0.488	0.696
chick	18.546	98	122	0.416	0.187
chicken	19.454	732	1,006	0.354	0.543
chip	16.858	120	154	0.277	0.341
chocolate	20.484	146	210	0.206	0.575
choo	18.811	404	453	0.544	0.424
chug	25.124	99	106	0.274	0.280
circle	16.745	225	260	0.372	0.517
circus	17.924	36	47	0.378	0.433
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
clam	20.313	44	49	0.733	0.240
clean	18.762	1,066	1,360	0.453	0.339
climb	20.483	122	155	0.342	0.410
clock	17.490	319	370	0.513	0.355
close	18.793	635	782	0.325	0.270
cloth	19.447	85	103	0.377	0.363
clothes	18.793	366	482	0.592	0.417
coffee	17.646	181	294	0.439	0.528
cold	21.310	660	861	0.224	0.309
color	16.649	322	376	0.380	0.205
comb	17.778	68	70	0.580	0.445
come	15.625	5,455	6,812	0.730	0.519
computer	20.712	169	241	0.324	0.381
cook	21.411	159	218	0.305	0.252
cookie	17.588	311	413	0.480	0.539
cool	15.716	466	536	0.336	0.279
couch	21.140	150	226	0.522	0.680
could	17.692	985	1,269	0.437	0.251
cow	16.045	1,014	1,056	0.859	0.408
crab	17.644	27	33	0.436	0.470
cracker	20.156	99	141	0.331	0.523
crayon	22.677	40	38	0.450	0.511
crazy	20.019	1,215	1,580	0.259	0.365
cream	20.180	1,056	1,373	0.507	0.303
crib	18.444	128	158	0.500	0.346
cry	17.148	598	723	0.410	0.421
cup	15.590	238	289	0.419	0.474
cut	17.490	252	354	0.283	0.347
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
cute	18.747	371	493	0.111	0.300
dad	9.486	27	31	0.627	0.593
dame	21.140	206	243	0.605	0.072
dark	18.059	177	203	0.606	0.147
[child name]	10.558	1,166	1,358	0.665	0.406
day	16.494	906	1,203	0.177	0.483
dear	21.081	154	202	0.209	0.225
deer	18.714	23	33	0.395	0.300
diamond	19.844	88	89	0.435	0.310
diaper	17.547	1,044	1,323	0.957	0.564
did	19.946	5,945	8,115	0.169	0.443
ding	20.942	127	153	0.280	0.137
dinner	21.418	754	1,025	0.397	0.351
dinosaur	19.512	82	94	0.498	0.440
dirty	18.715	253	335	0.472	0.297
dish	20.083	316	364	0.510	0.407
do	13.753	4,122	4,919	0.601	0.550
doctor	19.446	168	234	0.146	0.317
does	23.755	3,743	4,481	0.356	0.437
dog	16.058	1,405	1,476	0.701	0.428
doing	20.410	2,965	3,943	0.024	0.233
dolphin	21.411	133	155	0.372	0.405
done	11.642	327	420	0.293	0.348
donkey	19.420	26	31	0.097	0.455
door	16.784	158	191	0.447	0.195
dough	22.258	78	69	0.268	0.509
down	14.986	1,350	1,621	0.375	0.392
downstairs	19.682	316	462	0.348	0.366
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
draw	17.448	154	216	0.417	0.718
drink	19.844	866	1,208	0.269	0.490
driving	23.416	812	949	0.404	0.219
drum	17.548	127	156	0.457	0.243
dry	19.343	162	215	0.255	0.236
duck	11.276	79	81	0.703	0.237
dude	16.082	2,145	2,459	0.701	0.524
dump	17.957	65	93	0.186	0.489
eat	19.448	4,662	6,311	0.641	0.767
elephant	17.744	221	264	0.388	0.277
[sister name]	21.285	8	11	0.343	0.083
elmo	18.746	71	78	0.464	0.340
else	19.477	849	1,137	0.168	0.344
empty	19.356	141	183	0.344	0.361
end	20.441	392	553	0.276	0.341
engine	18.789	66	83	0.200	0.272
enough	21.523	1,056	1,380	0.261	0.209
eye	14.593	305	329	0.567	0.276
face	19.947	629	800	0.116	0.226
fall	16.422	297	373	0.273	0.130
fan	16.645	40	54	0.347	0.230
far	25.141	542	617	0.228	0.151
fast	20.044	421	520	0.342	0.244
feel	17.551	510	698	0.085	0.201
fell	20.422	547	687	0.154	0.272
find	19.347	1,321	1,628	0.191	0.236
fine	20.879	799	1,146	0.056	0.261
finger	18.844	157	196	0.201	0.326
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
finish	20.157	552	748	0.469	0.464
fire	17.953	312	386	0.388	0.417
firetruck	18.243	13	16	0.403	0.435
first	16.111	513	662	0.180	0.266
fish	9.608	66	92	0.727	0.448
five	17.560	1,688	2,056	0.315	0.456
fix	21.743	156	225	0.312	0.417
floor	14.986	113	149	0.322	0.312
flower	16.117	682	685	0.746	0.200
fly	17.980	252	313	0.416	0.479
fold	21.345	64	86	0.399	0.337
food	19.976	788	1,074	0.297	0.525
for	15.389	3,757	4,845	0.333	0.343
found	20.376	505	678	0.193	0.272
four	18.810	1,753	2,165	0.313	0.333
fox	18.481	179	193	0.763	0.353
fresh	21.708	165	241	0.228	0.173
friday	19.681	170	231	0.334	0.411
frog	16.578	663	704	0.609	0.357
from	10.660	141	162	0.346	0.299
full	20.984	1,103	1,349	0.497	0.425
fun	21.423	925	1,116	0.149	0.365
funny	17.978	425	556	0.149	0.213
garage	18.759	61	82	0.401	0.299
garbage	18.745	107	164	0.331	0.222
[name 2]	18.514	32	52	0.284	0.491
get	14.986	2,705	3,422	0.288	0.398
gimme	24.376	184	222	0.189	0.224
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
giraffe	18.359	139	152	0.396	0.361
girl	18.715	189	228	0.426	0.222
give	22.677	3,863	5,089	0.059	0.351
glasses	17.056	87	127	0.371	0.132
glider	22.451	19	35	0.276	0.461
go	14.985	6,104	7,439	0.570	0.486
god	19.421	698	978	0.172	0.335
gone	13.744	181	233	0.337	0.381
gonna	21.410	6,219	8,239	0.213	0.353
good	16.293	4,638	5,798	0.509	0.318
goodbye	17.678	200	234	0.410	0.564
goodness	22.344	676	817	0.294	0.254
goodnight	21.743	280	291	0.863	0.419
got	20.814	3,178	4,495	0.123	0.279
grape	18.457	90	118	0.387	0.585
gray	17.648	41	58	0.082	0.456
great	20.142	357	517	0.038	0.349
green	17.646	743	907	0.576	0.195
guava	18.349	21	31	0.285	0.594
gum	17.655	32	38	0.367	0.491
had	17.782	2,449	3,360	0.412	0.560
hair	17.512	265	304	0.383	0.272
hammer	21.789	32	42	0.390	0.779
hand	17.912	899	1,128	0.224	0.265
happened	21.178	1,175	1,589	0.119	0.334
happy	18.111	582	744	0.410	0.158
hard	20.813	679	928	0.128	0.238
has	24.379	2,291	2,782	0.049	0.284
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
hat	16.914	239	244	0.718	0.235
have	14.986	3,181	4,046	0.304	0.433
he	22.414	17,604	23,877	0.467	0.309
head	19.453	553	696	0.203	0.298
hear	24.060	1,133	1,295	0.189	0.136
heard	23.429	373	478	0.100	0.381
heart	16.848	208	228	0.478	0.250
helicopter	17.981	66	78	0.297	0.285
hello	16.758	1,177	1,430	0.335	0.310
help	17.723	641	871	0.268	0.205
her	22.415	2,590	3,234	0.360	0.226
here	13.584	3,422	4,106	0.621	0.360
hey	11.710	1,114	1,354	0.471	0.477
hi	12.662	917	1,112	0.565	0.424
hide	20.313	371	487	0.234	0.423
high	17.659	693	786	0.379	0.330
him	16.459	2,842	4,015	0.250	0.304
his	20.507	4,728	6,396	0.284	0.313
hit	20.341	226	315	0.210	0.436
hockey	19.347	9	17	0.172	0.496
hold	18.524	1,165	1,486	0.267	0.368
home	19.909	906	1,241	0.319	0.391
honey	19.678	287	379	0.210	0.360
hop	25.179	130	155	0.236	0.149
horse	18.812	849	946	0.594	0.246
hot	16.795	321	452	0.251	0.368
house	15.712	452	510	0.679	0.361
how	17.794	3,999	5,279	0.250	0.297
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
hug	19.914	279	361	0.414	0.345
hungry	16.328	576	722	0.377	0.350
hurt	22.388	611	739	0.149	0.298
i	10.959	2,683	3,307	0.770	0.519
ice	18.122	764	966	0.549	0.531
if	14.986	1,365	1,798	0.307	0.451
in	19.454	14,265	18,929	0.460	0.396
inside	19.976	450	585	0.117	0.204
is	13.185	5,495	6,210	0.881	0.494
it	11.725	3,975	4,963	0.497	0.449
jeans	21.312	40	37	0.403	0.448
jeep	25.212	18	12	0.450	0.503
job	17.446	1,497	1,716	0.501	0.398
joy	17.113	566	853	0.256	0.375
juice	16.688	347	473	0.604	0.782
jump	18.851	243	309	0.432	0.208
just	15.291	3,344	4,417	0.318	0.434
keep	19.353	686	934	0.047	0.292
key	17.718	116	148	0.291	0.177
kick	16.111	58	90	0.438	0.193
kid	20.151	610	842	0.176	0.357
kiss	19.410	815	964	0.434	0.338
kitchen	20.313	204	308	0.238	0.190
kite	20.873	50	62	0.474	0.609
know	15.578	3,891	5,061	0.287	0.428
lamp	20.916	55	59	0.547	0.367
lane	16.250	110	129	0.435	0.213
last	17.695	715	1,018	0.179	0.380
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
later	23.326	701	910	0.217	0.179
laundry	18.714	76	106	0.405	0.340
let	19.922	1,715	2,364	0	0.287
letters	22.415	125	135	0.337	0.193
lie	23.478	235	240	0.456	0.244
light	16.694	535	631	0.639	0.268
like	15.713	5,929	7,586	0.519	0.397
lion	18.261	204	255	0.187	0.299
listen	14.520	321	363	0.472	0.455
little	20.816	7,280	9,296	0.410	0.332
living	21.522	147	199	0.286	0.166
long	16.579	531	734	0.118	0.286
look	15.259	4,254	4,920	0.703	0.414
lots	21.016	427	522	0.164	0.223
love	20.410	1,742	2,163	0.290	0.390
mad	21.154	188	249	0.168	0.205
make	20.423	2,872	4,118	0.221	0.435
man	20.875	1,960	2,472	0.295	0.313
mango	16.494	392	491	0.671	0.784
many	23.512	1,956	2,292	0.316	0.088
matter	21.219	220	317	0.062	0.262
maybe	16.494	841	1,176	0.223	0.513
mcdonald	20.775	221	265	0.600	0.860
me	14.523	3,546	4,154	0.550	0.477
mean	19.976	1,325	1,894	0.215	0.373
medicine	19.611	259	362	0.329	0.428
meow	14.443	157	159	0.432	0.285
milk	16.527	966	1,274	0.468	0.565
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
mine	17.122	258	360	0.214	0.381
mix	20.979	212	261	0.300	0.500
mobile	23.275	16	18	0.399	0.232
mom	13.124	827	956	0.623	0.368
monday	20.245	151	192	0.293	0.380
money	19.679	125	169	0.239	0.419
monkey	16.580	1,050	1,252	0.401	0.443
moo	13.517	158	167	0.646	1
moon	15.175	428	410	0.813	0.731
moose	24.278	21	11	0.320	0.498
more	13.159	1,231	1,449	0.434	0.484
morning	16.121	497	647	0.337	0.366
mouse	17.678	740	791	0.612	0.360
mouth	18.110	1,514	1,859	0.596	0.531
move	19.976	710	913	0.124	0.343
much	17.657	1,268	1,782	0.198	0.356
music	21.051	468	551	0.447	0.442
my	13.290	1,275	1,513	0.408	0.319
nap	24.084	362	445	0.231	0.242
neat	19.919	94	113	0.235	0.462
need	21.889	2,230	3,017	0.029	0.293
neigh	15.755	121	129	0.285	0.359
nemo	18.579	95	112	0.408	0.626
new	20.441	1,101	1,481	0.254	0.381
next	11.743	120	145	0.296	0.307
nice	19.678	3,009	3,830	0.196	0.413
nicely	21.360	636	723	0.267	0.212
night	10.862	90	94	0.546	0.487
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
nine	23.756	875	1,014	0.302	0.229
no	11.326	1,671	1,977	0.597	0.381
nose	18.146	661	747	0.466	0.297
not	14.986	3,467	4,157	0.585	0.224
now	19.920	5,746	7,772	0.112	0.344
number	21.523	947	1,134	0.603	0.491
octopus	19.543	124	135	0.577	0.508
of	19.946	10,327	13,853	0.428	0.438
off	16.577	1,007	1,312	0.291	0.378
oh	9.955	360	458	0.474	0.400
oil	19.447	174	241	0.369	0.378
ok	9.952	439	546	0.508	0.357
old	19.393	1,215	1,481	0.680	0.879
on	10.314	1,130	1,299	0.739	0.375
one	14.710	4,684	5,397	0.620	0.326
only	15.748	486	639	0.193	0.405
open	16.480	904	1,098	0.470	0.246
or	16.655	2,551	3,450	0.279	0.345
orange	16.913	376	500	0.350	0.390
other	19.093	1,490	2,031	0.092	0.415
ouch	16.795	71	84	0.236	0.208
our	21.708	1,490	1,853	0.216	0.271
out	15.056	2,012	2,393	0.401	0.509
outside	19.309	531	727	0.364	0.303
over	21.052	2,367	3,018	0.283	0.469
owl	17.744	108	137	0.356	0.187
pajamas	19.455	63	77	0.385	0.296
pancakes	21.975	69	82	0.358	0.548
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
panda	20.313	110	118	0.523	0.225
pants	16.861	335	392	0.607	0.261
papa	16.912	144	144	0.900	0.248
paper	20.716	202	287	0.225	0.373
park	19.943	95	123	0.278	0.307
party	21.654	164	228	0.209	0.232
pasta	20.181	197	268	0.470	0.490
pea	17.892	327	446	0.666	0.618
pear	19.392	218	293	0.471	0.725
pee	17.513	388	437	0.952	0.554
peek	16.179	93	116	0.386	0.224
pen	16.987	323	447	0.534	0.909
people	25.186	1,321	1,594	0.240	0.326
phone	16.625	328	436	0.236	0.425
pick	19.309	626	823	0.143	0.241
picture	18.445	566	692	0.294	0.403
pie	17.877	273	333	0.675	0.395
piece	22.258	523	714	0.484	0.464
pig	18.146	1,202	1,456	0.437	0.578
pillow	20.441	118	141	0.570	0.273
pink	17.493	158	163	0.396	0.364
pizza	20.156	140	239	0.413	0.545
plane	17.460	137	197	0.395	0.266
plate	21.975	108	169	0.365	0.477
play	19.145	2,712	3,685	0.263	0.455
please	16.456	596	733	0.282	0.576
plum	19.456	104	126	0.388	0.265
police	19.643	144	157	0.493	0.340
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
poop	17.481	625	803	0.604	0.385
pop	17.556	246	279	0.365	0.294
potato	18.747	207	274	0.464	0.534
press	21.775	1,446	1,597	0.928	0.325
pretty	20.411	1,009	1,316	0.131	0.284
prince	20.814	66	95	0.380	0.343
pull	23.310	594	683	0.280	0.187
puppy	17.714	91	98	0.475	0.255
purple	16.795	238	240	0.412	0.357
push	16.694	521	658	0.412	0.465
put	20.376	6,320	8,705	0.302	0.341
puzzle	15.456	17	25	0.332	0.333
race	20.388	225	284	0.512	0.460
racecar	23.873	13	18	0.152	0.538
rain	19.448	571	619	0.537	0.313
rainbow	23.923	233	229	0.439	0.143
raining	19.448	571	619	0.537	0.313
read	22.142	1,447	1,813	0.413	0.443
ready	18.853	1,768	2,283	0.302	0.366
really	21.178	2,881	3,940	0.173	0.294
red	18.412	1,369	1,627	0.323	0.209
remember	20.877	855	1,178	0.152	0.327
rice	19.924	185	293	0.507	0.498
ride	25.186	244	242	0.271	0.297
right	16.194	4,460	5,641	0.328	0.389
robot	20.153	55	66	0.326	0.464
rock	17.659	112	144	0.389	0.216
room	20.154	509	705	0.293	0.220
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
round	20.942	1,599	1,882	0.879	0.623
run	18.714	460	618	0.291	0.158
[mother name]	17.525	654	1,063	0.282	0.351
said	16.645	1,461	1,815	0.475	0.202
salad	23.324	143	224	0.376	0.314
sandals	19.946	6	11	0.371	0.527
sandwich	23.809	100	135	0.269	0.450
sara	23.414	47	56	0.382	0.361
saw	23.761	1,003	1,129	0.285	0.190
say	20.849	7,757	9,666	0.436	0.342
school	19.688	225	279	0.329	0.377
sea	18.812	387	417	0.974	0.357
seat	22.112	165	228	0.307	0.308
see	19.356	7,764	9,950	0.434	0.360
set	21.912	389	528	0.313	0.322
seven	17.691	709	860	0.186	0.417
shake	21.140	140	172	0.358	0.330
shark	18.010	48	69	0.368	0.303
she	20.845	2,847	3,973	0.440	0.358
sheep	15.389	954	1,042	0.624	0.407
shirt	16.882	240	325	0.331	0.355
shoe	16.624	274	354	0.379	0.343
should	14.986	716	897	0.193	0.311
show	19.481	2,128	2,619	0.339	0.280
shower	18.910	210	301	0.438	0.324
side	20.353	423	605	0.222	0.336
silver	20.190	70	77	0.438	0.360
sing	19.679	1,266	1,460	0.574	0.449
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
sir	20.721	914	1,074	0.627	0.455
sit	18.812	1,817	2,474	0.280	0.325
six	22.756	1,237	1,580	0.264	0.511
skin	21.078	98	131	0.315	0.207
sky	17.678	691	754	0.702	0.462
sleep	17.597	1,243	1,534	0.712	0.335
small	10.961	53	64	0.437	0.370
snow	18.910	178	237	0.353	0.156
so	16.127	4,772	6,259	0.267	0.330
soap	21.683	41	46	0.382	0.215
soccer	20.044	38	48	0.397	0.452
socks	17.512	334	450	0.552	0.272
some	18.361	4,514	6,242	0.485	0.394
something	15.760	944	1,256	0.077	0.395
song	22.211	572	646	0.342	0.176
sorry	16.421	348	475	0.172	0.372
soup	21.314	260	316	0.430	0.617
spider	22.616	1,067	1,092	0.613	0.430
spoon	16.693	422	524	0.705	0.812
squirrel	19.254	41	45	0.456	0.225
stairs	19.682	31	44	0.091	0.243
stand	20.423	726	890	0.538	0.341
star	13.111	176	187	0.773	0.358
starfish	19.676	46	51	0.518	0.299
stay	20.388	525	707	0.257	0.336
stick	20.879	361	534	0.409	0.369
stop	21.912	1,543	1,885	0.239	0.293
store	20.710	281	389	0.311	0.289
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
straw	19.445	59	77	0.323	0.557
strawberry	18.779	59	92	0.444	0.624
stuck	17.560	131	166	0.272	0.213
stuff	17.456	655	947	0.186	0.366
sugar	18.753	239	372	0.536	0.513
sun	14.986	517	560	0.678	0.509
sure	21.778	1,319	1,875	0.155	0.310
sweet	22.976	634	776	0.366	0.341
swimming	19.687	77	86	0.420	0.305
table	18.522	422	556	0.446	0.468
tail	20.352	198	206	0.565	0.161
take	20.108	3,105	4,421	0.173	0.327
talk	20.350	655	844	0.184	0.389
taste	20.719	510	702	0.612	0.700
taxi	18.344	30	35	0.407	0.245
tea	17.714	170	275	0.473	0.564
teddy	20.376	122	142	0.462	0.269
teeth	20.845	530	648	0.257	0.283
telephone	19.923	240	256	0.678	0.238
tell	17.912	790	1,035	0.069	0.259
ten	19.145	742	1,006	0.320	0.577
thank	15.647	449	531	0.253	0.450
that	14.515	7,863	9,650	0.465	0.479
the	10.314	2,883	3,101	1	0.476
them	14.986	1,257	1,433	0.903	0.281
then	14.986	1,665	2,077	0.363	0.368
there	16.113	4,868	5,855	0.469	0.247
these	23.289	2,174	2,732	0.065	0.190
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
they	20.146	4,196	5,497	0.468	0.350
thing	22.744	4,526	6,078	0.229	0.267
think	20.441	4,798	6,934	0.340	0.389
this	14.445	6,034	7,208	0.689	0.374
thomas	21.352	54	38	0.483	0.634
those	22.909	1,967	2,585	0.149	0.131
though	22.760	1,037	1,407	0.082	0.316
three	16.191	2,083	2,366	0.515	0.388
through	16.421	516	598	0.412	0.264
throw	16.857	828	1,220	0.379	0.556
thumper	20.153	67	94	0.404	0.652
tickle	18.662	195	227	0.427	0.176
tiger	18.714	63	76	0.346	0.341
time	23.379	4,424	5,696	0.080	0.195
tiny	20.355	230	274	0.499	0.335
tired	20.441	570	727	0.201	0.323
to	13.876	7,304	8,944	0.566	0.460
today	17.648	1,588	2,362	0.151	0.439
toe	16.861	130	161	0.306	0.329
toes	16.861	130	161	0.306	0.329
together	24.075	680	817	0.178	0.315
tomorrow	16.527	305	462	0.281	0.476
tongue	17.547	187	207	0.410	0.208
too	20.376	2,913	4,043	0.107	0.560
toothbrush	18.458	35	47	0.407	0.274
toothpaste	20.720	46	66	0.400	0.690
top	22.677	455	589	0.174	0.292
touch	17.981	369	465	0.217	0.234
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
towel	18.945	78	102	0.277	0.250
town	14.986	88	92	0.405	0.273
toy	19.348	713	983	0.334	0.360
track	17.514	86	104	0.300	0.326
tractor	19.387	84	98	0.440	0.480
train	15.546	408	427	0.676	0.323
tree	16.813	467	519	0.623	0.323
triangle	19.177	298	320	0.360	0.359
[name 3]	19.478	85	116	0.363	0.340
trouble	20.019	176	225	0.232	0.187
truck	14.811	732	854	0.722	0.593
true	14.175	49	58	0.300	0.180
trunk	17.687	24	30	0.382	0.054
try	20.815	3,018	4,075	0.339	0.322
tummy	19.909	90	113	0.338	0.651
tunnel	18.910	10	11	0.142	0.451
turn	23.287	2,269	2,550	0.664	0.315
turtle	18.386	302	311	0.742	0.466
tweet	20.341	48	60	0.238	0.594
twinkle	19.946	224	226	0.346	0.310
two	16.480	2,460	2,933	0.393	0.241
under	20.376	379	497	0.314	0.349
up	13.756	2,619	3,123	0.501	0.477
vaseline	20.179	67	91	0.608	0.356
very	20.978	2,005	2,806	0.099	0.352
vroom	15.490	135	155	0.298	0.337
wait	11.726	250	281	0.490	0.245
walk	18.679	968	1,297	0.479	0.374
Continued on next page					

word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
walrus	21.523	19	20	0.305	0.340
want	13.060	3,523	4,297	0.582	0.529
was	16.456	4,251	5,827	0.446	0.291
wash	19.354	341	470	0.389	0.350
watch	19.390	747	1,026	0.144	0.351
water	13.049	791	941	0.581	0.424
way	22.141	2,096	2,700	0.133	0.283
we	20.157	12,197	16,672	0.176	0.421
wear	21.912	422	493	0.652	0.420
well	17.658	1,424	2,069	0.202	0.383
were	20.157	12,197	16,672	0.176	0.421
wet	21.176	381	461	0.385	0.448
what	10.650	1,180	1,414	0.651	0.372
wheel	17.688	900	1,021	0.823	0.795
when	24.054	4,919	6,283	0.149	0.236
where	13.556	2,869	3,211	0.823	0.393
which	15.278	612	742	0.369	0.402
whine	17.457	53	60	0.396	0.418
whistle	21.342	109	127	0.449	0.294
white	17.695	375	462	0.404	0.217
who	20.391	2,996	3,823	0.292	0.359
why	16.524	1,821	2,367	0.176	0.256
will	16.160	1,044	1,314	0.558	0.455
windmill	22.249	11	26	0.214	0.523
window	19.446	223	272	0.422	0.367
wipe	21.708	288	381	0.257	0.321
with	20.341	7,689	10,402	0.331	0.381
wonder	20.084	319	389	0.241	0.268
Continued on next page					

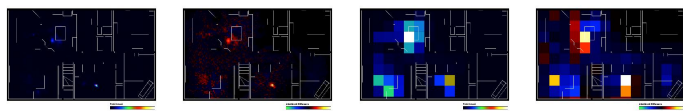
word	AoA	Utterances	Points	KL(P,Q)	RK(Q)
woof	17.980	296	321	0.472	0.246
wool	20.984	712	819	0.586	0.252
work	20.815	1,325	1,863	0.304	0.436
wormy	21.654	48	47	0.418	0.352
would	17.561	1,196	1,647	0.459	0.371
wow	15.154	1,440	1,712	0.393	0.415
wrong	18.745	692	914	0.078	0.321
yellow	18.061	869	1,013	0.360	0.381
yes	16.123	14,555	19,190	0.417	0.338
yet	15.248	193	243	0.155	0.227
yogurt	17.588	610	872	0.490	0.844
you	12.721	14,524	17,466	0.805	0.466
yuck	16.728	289	360	0.609	0.383
yum	18.661	1,147	1,450	0.750	0.891
zoo	16.791	119	139	0.717	0.339
zoom	21.541	104	154	0.238	0.422

Appendix B

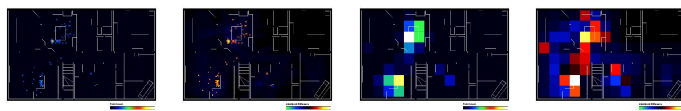
Visualizations for Target Words

This appendix provides visualizations for each of the 658 words in the child's vocabulary before the age of two. Visualizations are all based on the learning period of the word - that is, the period before the child's first production of the word. Each entry displays the number of utterances the word appeared in, the age of acquisition in months, an icon (in green - darker is lower value) denoting the relative value of Ripley's K of the spatial distribution for the word, an icon (in blue - darker is lower value) denoting the relative value of KL-divergence of the spatial distribution for the word, and four visualizations: (1) heat map with 100mm bins; (2) difference map with 100mm bins; (3) heat map with 1000mm bins; (4) difference map with 1000mm bins.

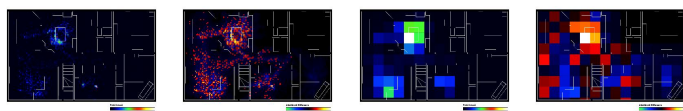
"a" Utterances: 13,566 AoA: 15.6



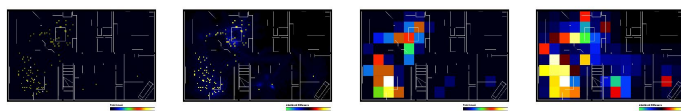
"aboard" Utterances: 69 AoA: 17.6



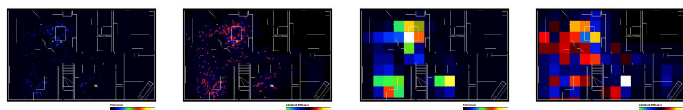
"about" Utterances: 1,439 AoA: 16.5



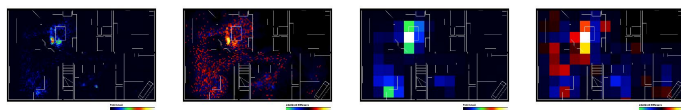
"accident" Utterances: 76 AoA: 20.4



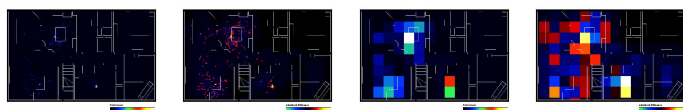
"after" Utterances: 293 AoA: 15.5



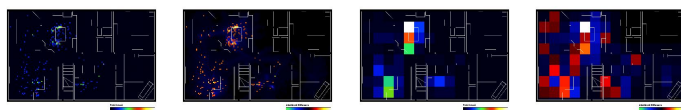
"again" Utterances: 2,566 AoA: 20.3



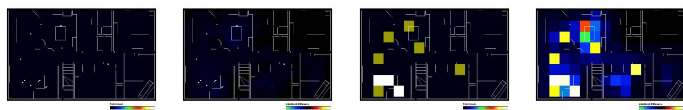
"air" Utterances: 192 AoA: 21.7



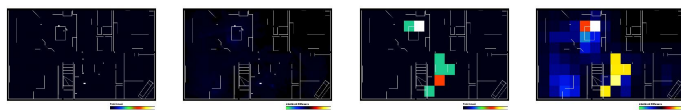
"airplane" Utterances: 170 AoA: 17.5



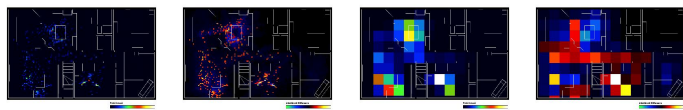
"album" Utterances: 11 AoA: 20.8



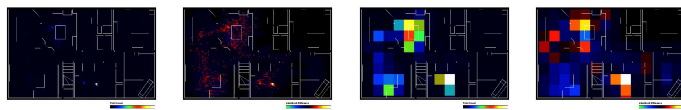
"[name 1]" Utterances: 6 AoA: 20.0



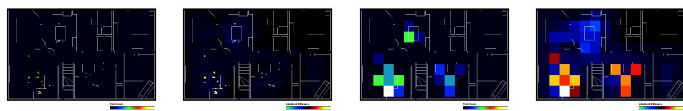
"[nanny name]" Utterances: 407 AoA: 15.5



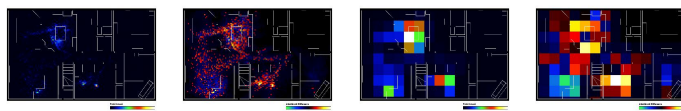
"all" Utterances: 994 AoA: 11.6



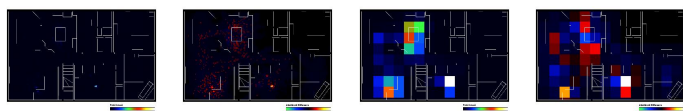
"alligator" Utterances: 36 AoA: 18.2



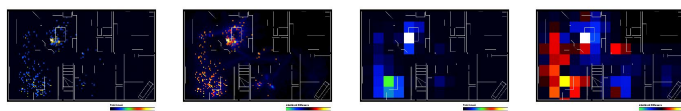
"alright" Utterances: 1,841 AoA: 19.4



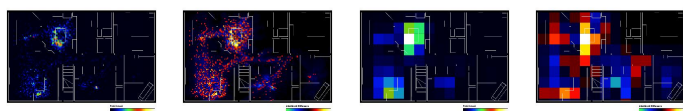
"am" Utterances: 698 AoA: 20.3



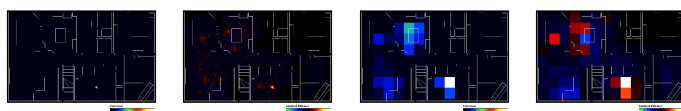
"ambulance" Utterances: 324 AoA: 21.5



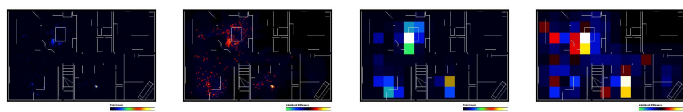
"an" Utterances: 1,830 AoA: 23.8



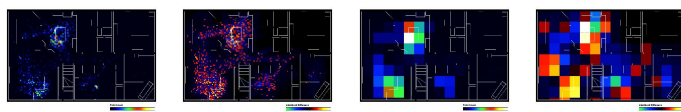
"and" Utterances: 2,998 AoA: 11.0



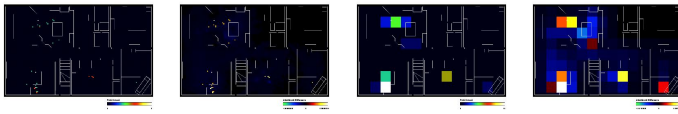
"animal" Utterances: 373 AoA: 19.5



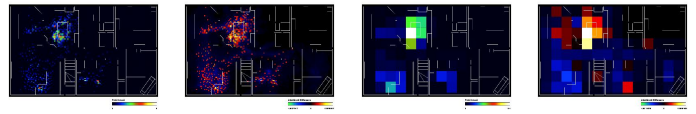
"another" Utterances: 1,237 AoA: 22.7



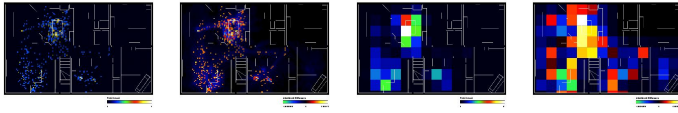
"ant" Utterances: 34 AoA: 23.0



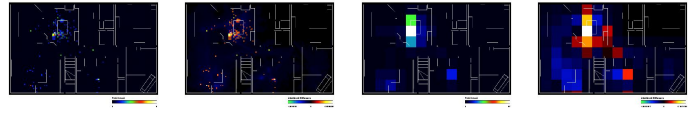
"any" Utterances: 796 AoA: 16.1



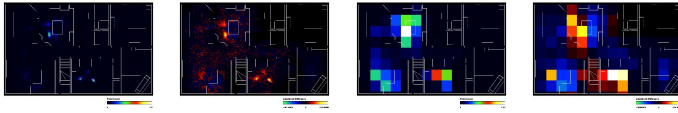
"anything" Utterances: 462 AoA: 16.1



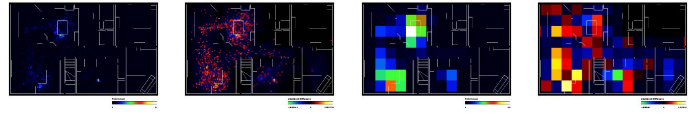
"apple" Utterances: 155 AoA: 15.3



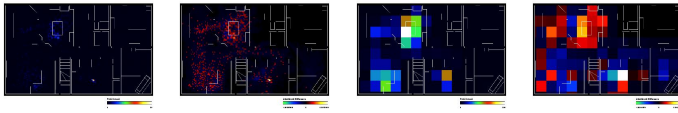
"are" Utterances: 6,846 AoA: 15.0



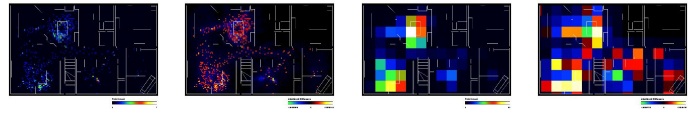
"around" Utterances: 1,108 AoA: 20.7



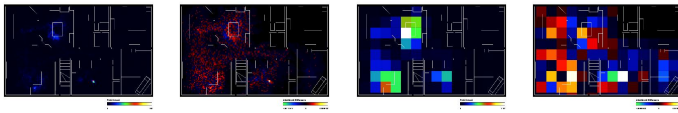
"as" Utterances: 1,134 AoA: 18.5



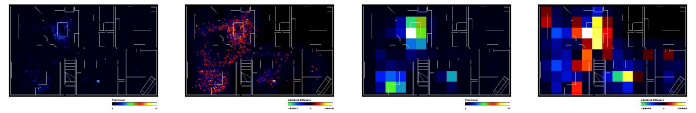
"ask" Utterances: 715 AoA: 23.9



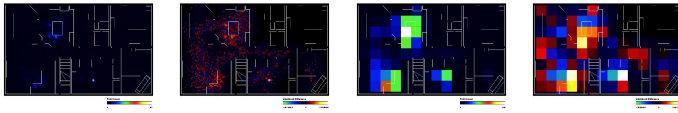
"at" Utterances: 6,736 AoA: 21.7



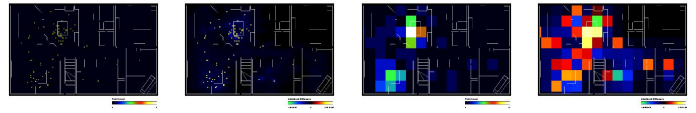
"ate" Utterances: 888 AoA: 25.5



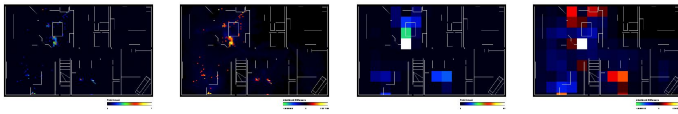
"away" Utterances: 2,218 AoA: 22.9



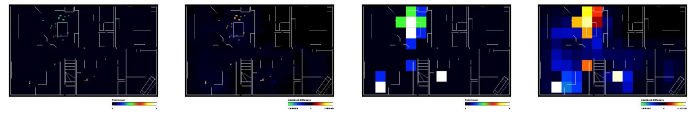
"awesome" Utterances: 61 AoA: 21.2



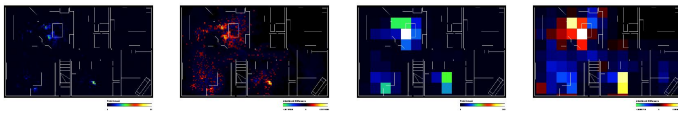
"baa" Utterances: 176 AoA: 11.1



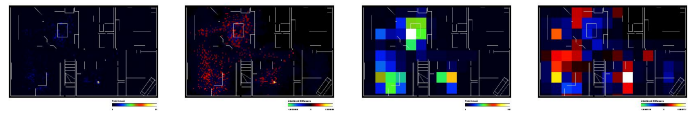
"baba" Utterances: 24 AoA: 15.0



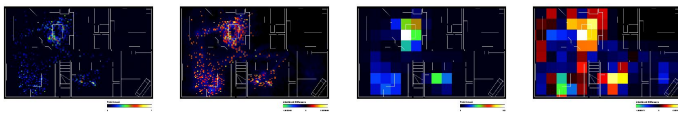
"baby" Utterances: 1,524 AoA: 15.8



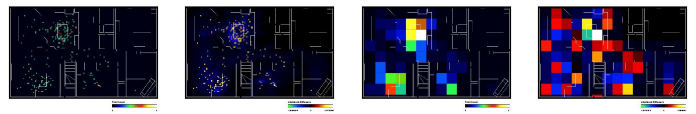
"back" Utterances: 887 AoA: 14.5



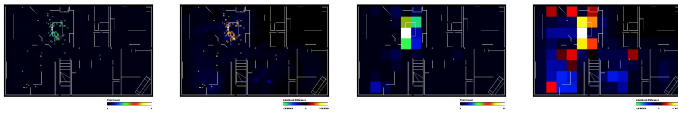
"bad" Utterances: 598 AoA: 17.6



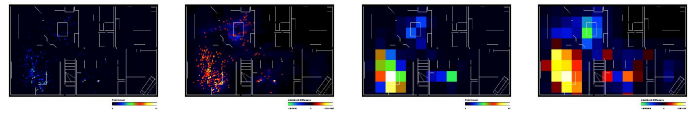
"bag" Utterances: 161 AoA: 17.7



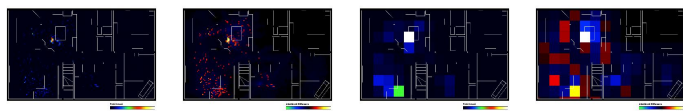
"bagel" Utterances: 48 AoA: 21.3



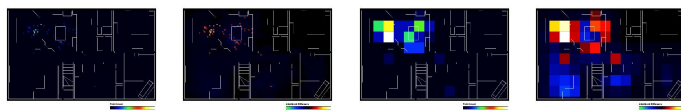
"ball" Utterances: 411 AoA: 12.9



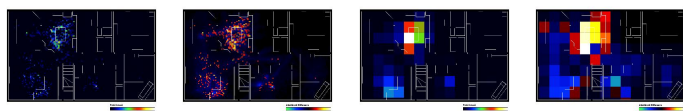
"balloon" Utterances: 287 AoA: 17.7



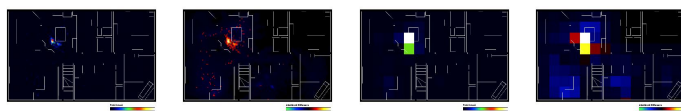
"bambi" Utterances: 51 AoA: 18.6



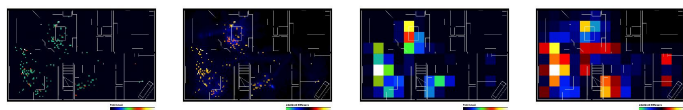
"banana" Utterances: 490 AoA: 20.3



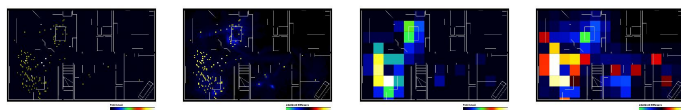
"barney" Utterances: 335 AoA: 18.9



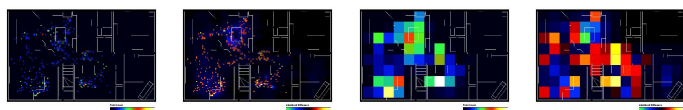
"basket" Utterances: 165 AoA: 20.2



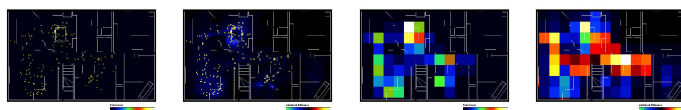
"basketball" Utterances: 94 AoA: 20.9



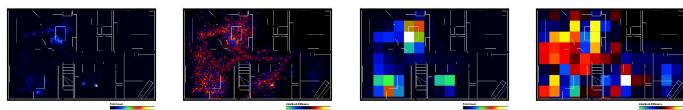
"bath" Utterances: 375 AoA: 16.6



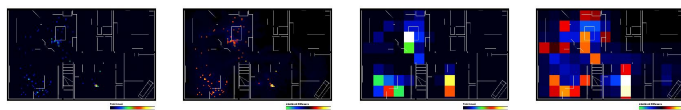
"bathroom" Utterances: 124 AoA: 18.8



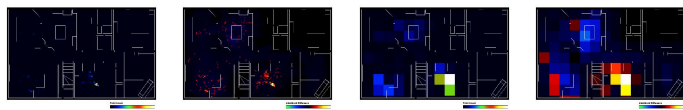
"be" Utterances: 3,538 AoA: 16.9



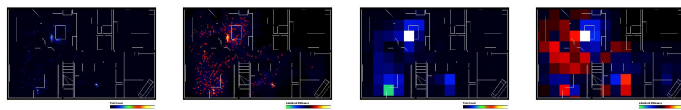
"beach" Utterances: 133 AoA: 23.5



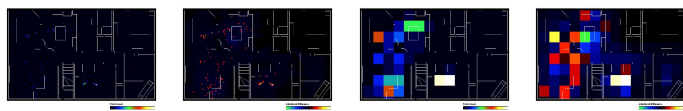
"bear" Utterances: 430 AoA: 14.6



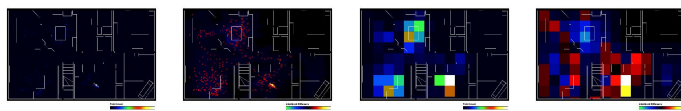
"beautiful" Utterances: 567 AoA: 21.2



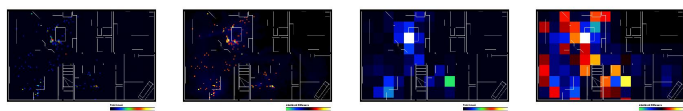
"because" Utterances: 88 AoA: 10.6



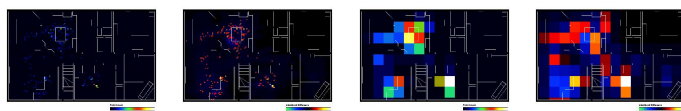
"bed" Utterances: 454 AoA: 18.5



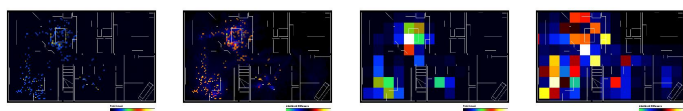
"bee" Utterances: 181 AoA: 17.6



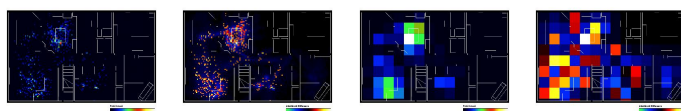
"been" Utterances: 186 AoA: 11.3



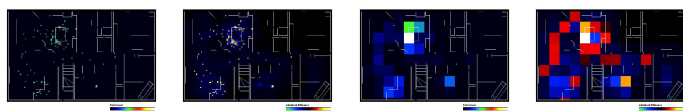
"beep" Utterances: 236 AoA: 22.4



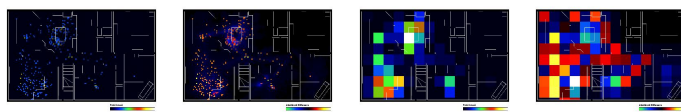
"before" Utterances: 501 AoA: 16.4



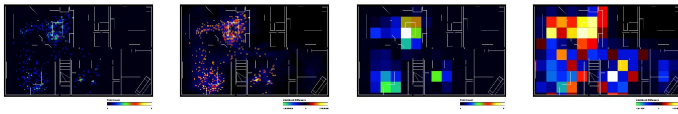
"beginning" Utterances: 92 AoA: 20.1



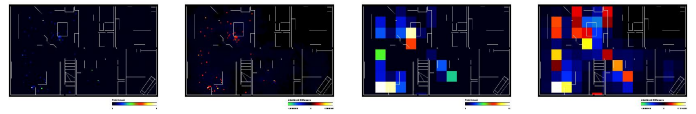
"behind" Utterances: 279 AoA: 24.9



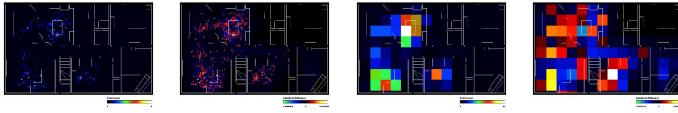
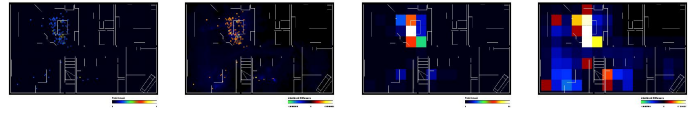
"being" Utterances: 467 AoA: 19.9



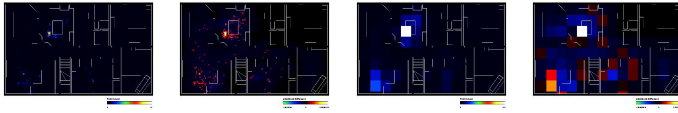
"bell" Utterances: 80 AoA: 16.9



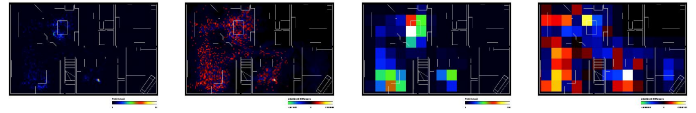
"better" Utterances: 391 AoA: 15.7

**"bib" Utterances: 91 AoA: 19.9**

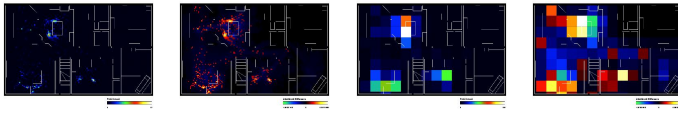
"bicycle" Utterances: 272 AoA: 18.5



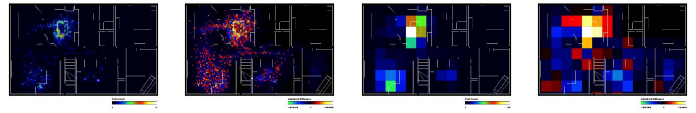
"big" Utterances: 1,936 AoA: 17.5



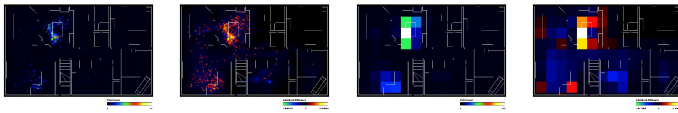
"bird" Utterances: 779 AoA: 16.7



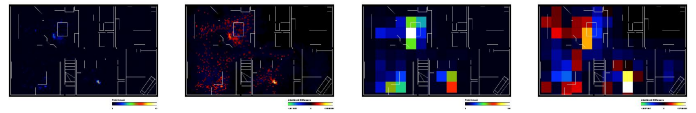
"bit" Utterances: 1,373 AoA: 21.1



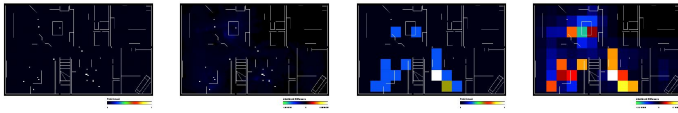
"bite" Utterances: 574 AoA: 20.8



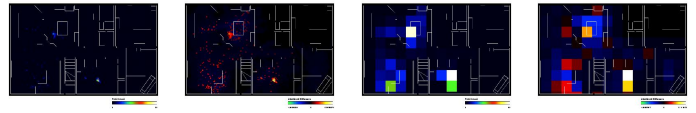
"black" Utterances: 1,120 AoA: 18.0



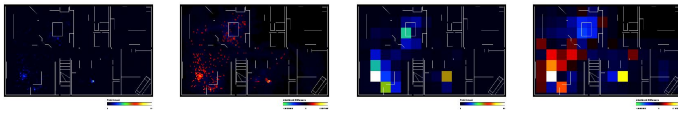
"blanket" Utterances: 16 AoA: 13.2



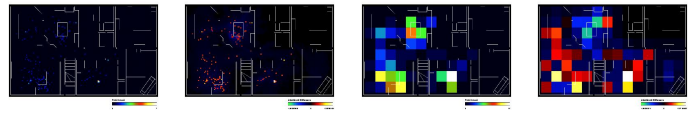
"blue" Utterances: 463 AoA: 16.0



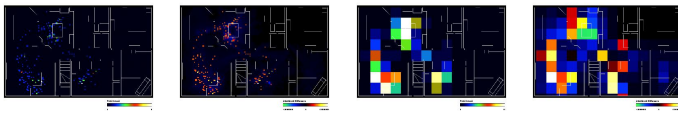
"boat" Utterances: 351 AoA: 16.8



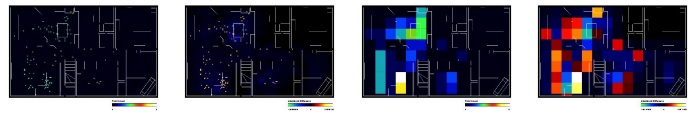
"body" Utterances: 143 AoA: 21.0



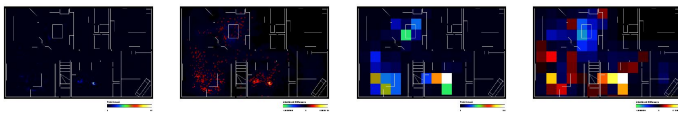
"boo" Utterances: 182 AoA: 15.5



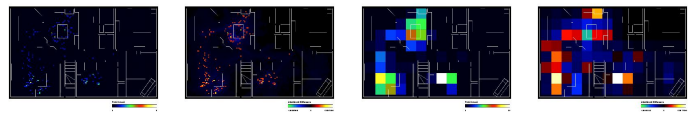
"booger" Utterances: 73 AoA: 17.7



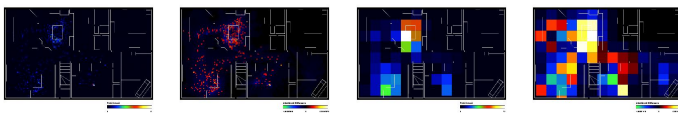
"book" Utterances: 716 AoA: 15.0



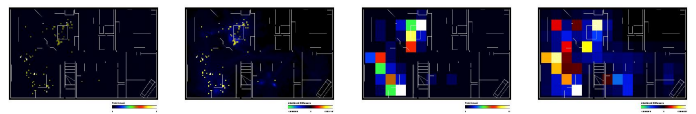
"boom" Utterances: 166 AoA: 16.2

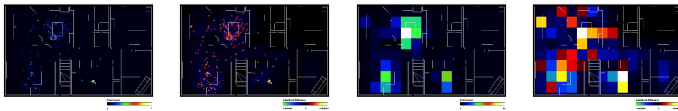


"bottle" Utterances: 374 AoA: 19.6

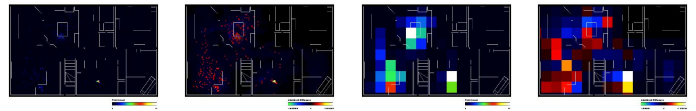


"bounce" Utterances: 70 AoA: 19.5

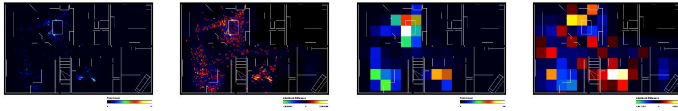


"bowl" Utterances: 255 AoA: 23.0

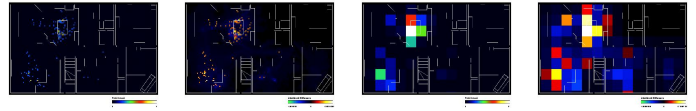
"box" Utterances: 297 AoA: 19.4



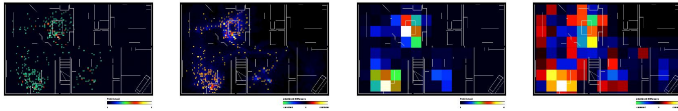
"boy" Utterances: 914 AoA: 14.8



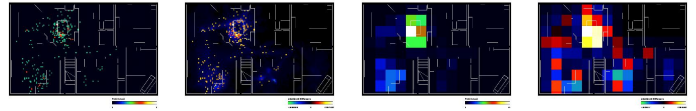
"bread" Utterances: 106 AoA: 15.0



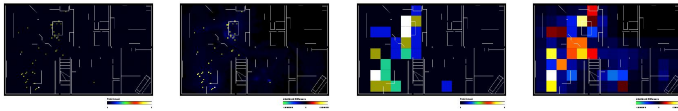
"break" Utterances: 306 AoA: 19.4



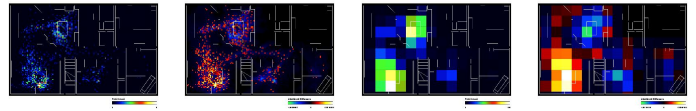
"breakfast" Utterances: 176 AoA: 19.0



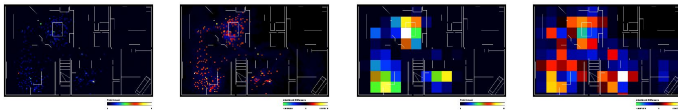
"bridge" Utterances: 38 AoA: 19.6



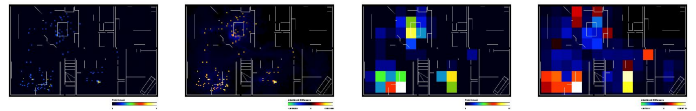
"bring" Utterances: 1,189 AoA: 23.2



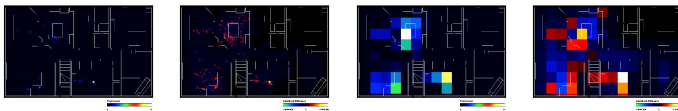
"broke" Utterances: 249 AoA: 20.4



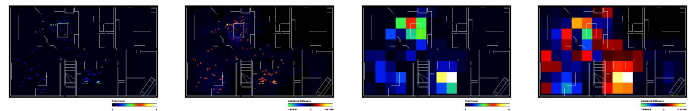
"brother" Utterances: 106 AoA: 19.7



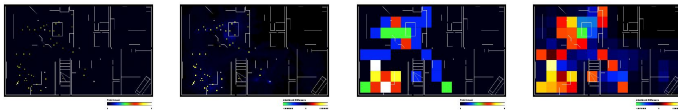
"brown" Utterances: 256 AoA: 16.7



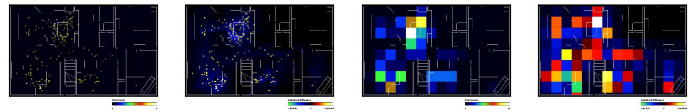
"brush" Utterances: 140 AoA: 16.4



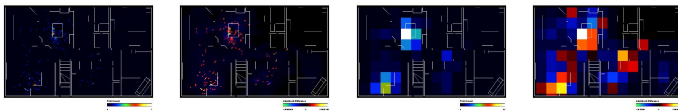
"bubble" Utterances: 48 AoA: 15.2



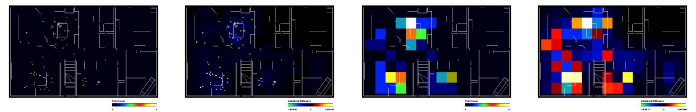
"buddy" Utterances: 136 AoA: 21.2



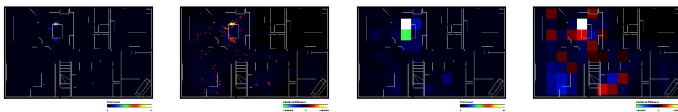
"bug" Utterances: 145 AoA: 17.0



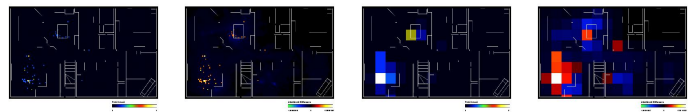
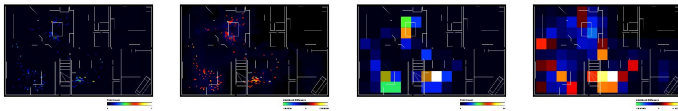
"bum" Utterances: 64 AoA: 17.6



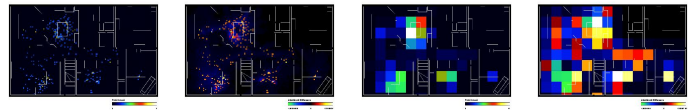
"bump" Utterances: 139 AoA: 16.8



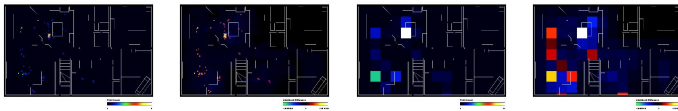
"bun" Utterances: 60 AoA: 16.6

**"bunny" Utterances: 220 AoA: 18.6**

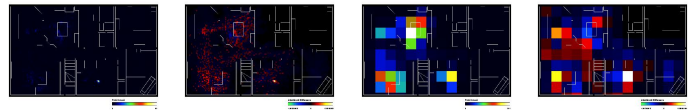
"burp" Utterances: 247 AoA: 24.8



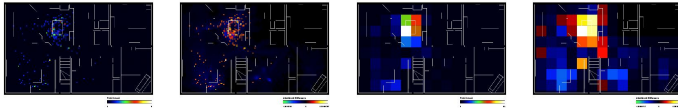
"bus" Utterances: 76 AoA: 14.7



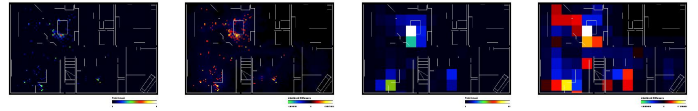
"but" Utterances: 1,785 AoA: 14.5



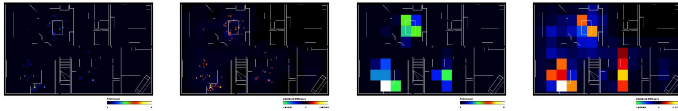
"butter" Utterances: 184 AoA: 23.3



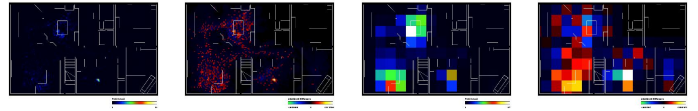
"butterfly" Utterances: 256 AoA: 18.7



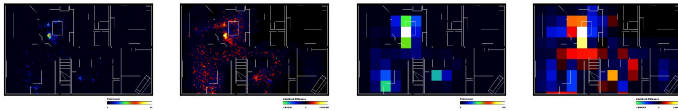
"button" Utterances: 97 AoA: 13.3



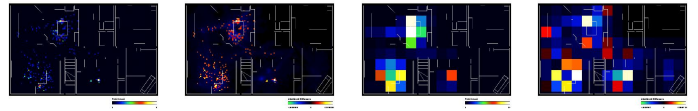
"by" Utterances: 1,061 AoA: 19.9



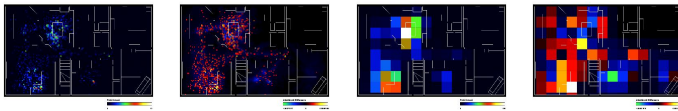
"bye" Utterances: 1,049 AoA: 15.0



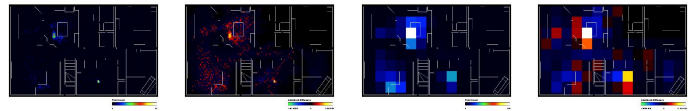
"cake" Utterances: 269 AoA: 20.8



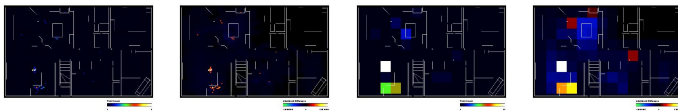
"call" Utterances: 1,031 AoA: 21.4



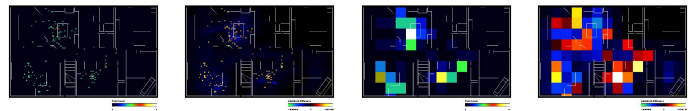
"came" Utterances: 1,203 AoA: 19.6



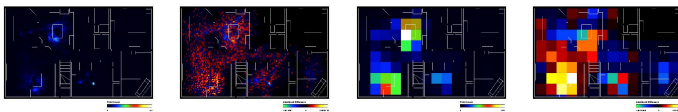
"camel" Utterances: 62 AoA: 18.6



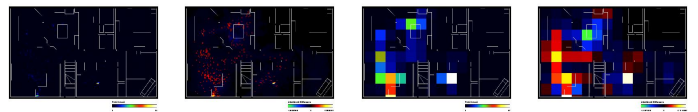
"camera" Utterances: 102 AoA: 16.2



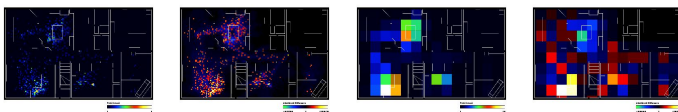
"can" Utterances: 9,810 AoA: 20.9



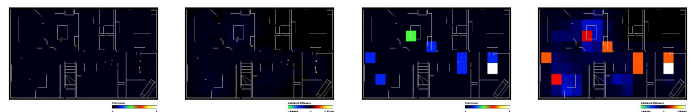
"car" Utterances: 479 AoA: 12.9



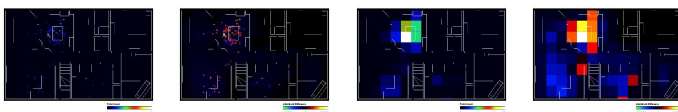
"careful" Utterances: 687 AoA: 20.2



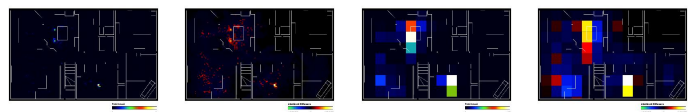
"carpet" Utterances: 11 AoA: 20.9



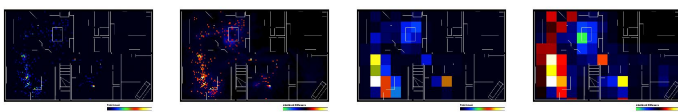
"carrot" Utterances: 101 AoA: 18.7



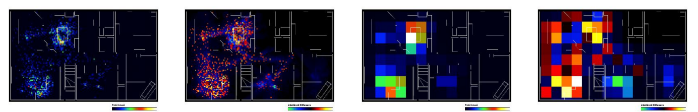
"cat" Utterances: 667 AoA: 14.7



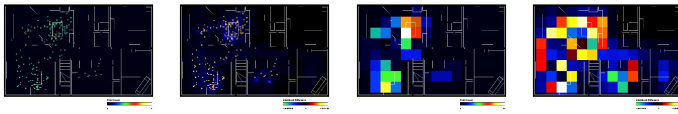
"catch" Utterances: 248 AoA: 18.5



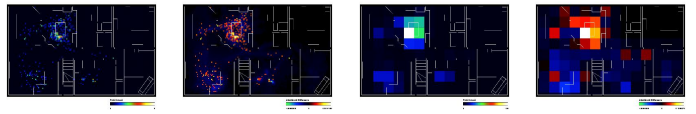
"cause" Utterances: 1,168 AoA: 19.9



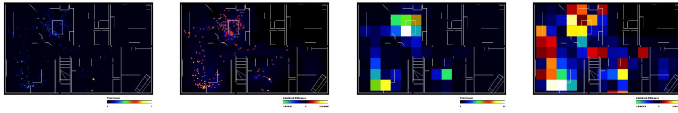
"cell" Utterances: 136 AoA: 21.9



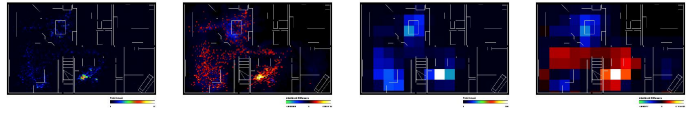
"cereal" Utterances: 328 AoA: 19.4



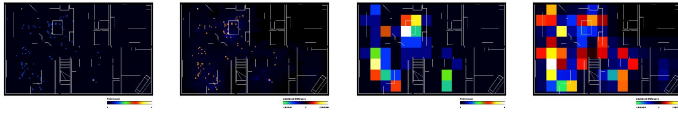
"chair" Utterances: 213 AoA: 15.0



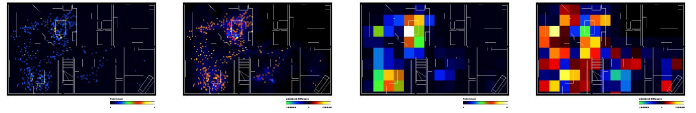
"change" Utterances: 1,204 AoA: 18.8



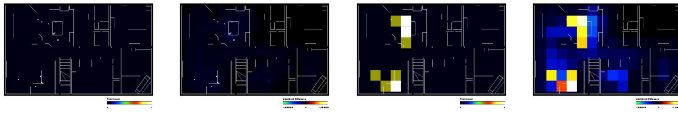
"chase" Utterances: 65 AoA: 19.7



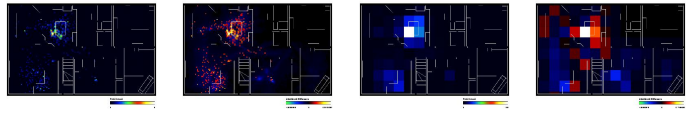
"check" Utterances: 382 AoA: 20.3



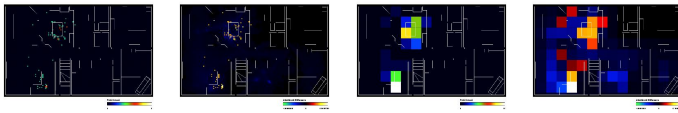
"cheerios" Utterances: 24 AoA: 21.8



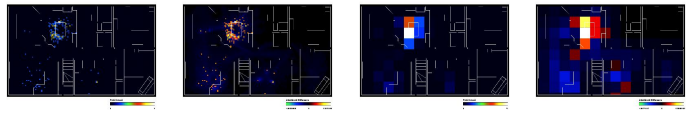
"cheese" Utterances: 429 AoA: 19.5



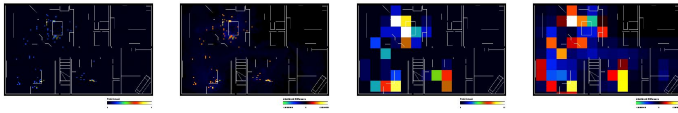
"cherries" Utterances: 54 AoA: 21.7



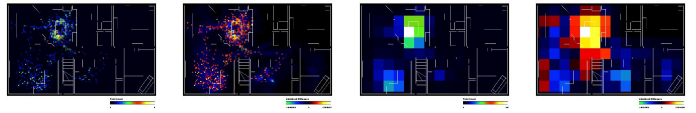
"chew" Utterances: 177 AoA: 17.7



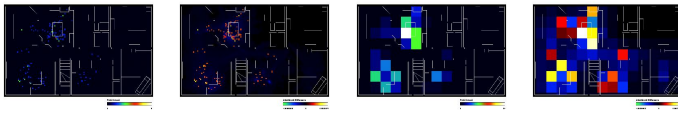
"chick" Utterances: 98 AoA: 18.5



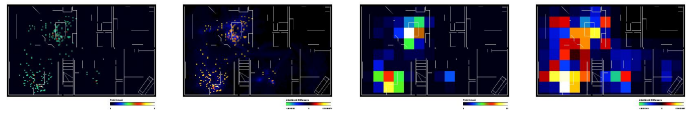
"chicken" Utterances: 732 AoA: 19.5



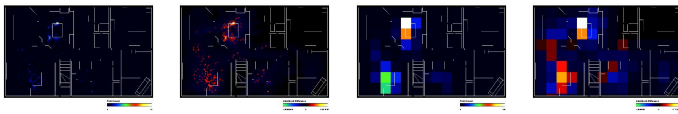
"chip" Utterances: 120 AoA: 16.9



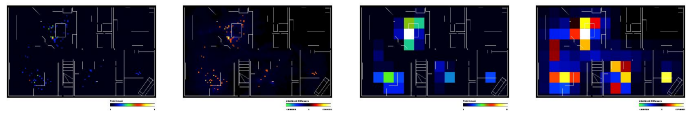
"chocolate" Utterances: 146 AoA: 20.5



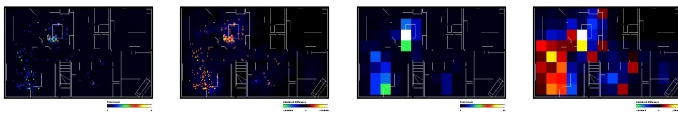
"choo" Utterances: 404 AoA: 18.8



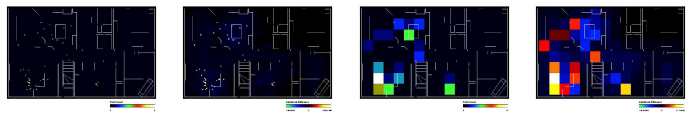
"chug" Utterances: 99 AoA: 25.1



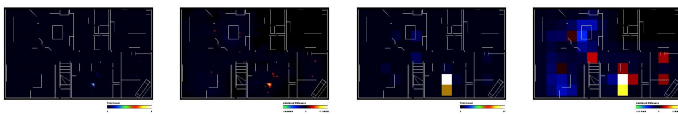
"circle" Utterances: 225 AoA: 16.7



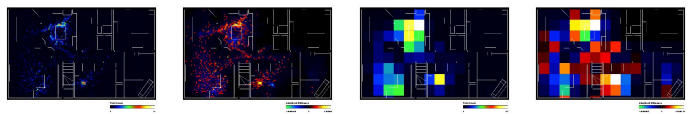
"circus" Utterances: 36 AoA: 17.9



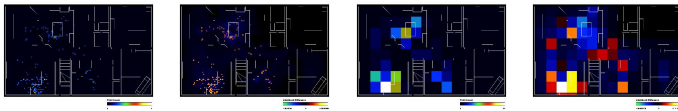
"clam" Utterances: 44 AoA: 20.3



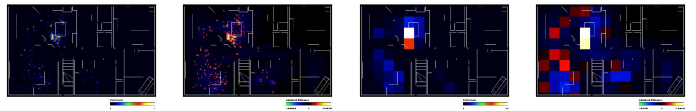
"clean" Utterances: 1,066 AoA: 18.8



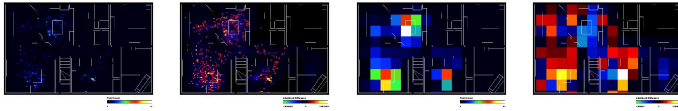
"climb" Utterances: 122 AoA: 20.5



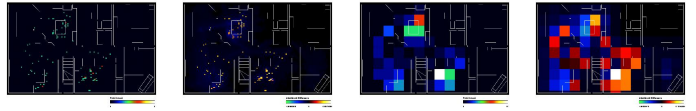
"clock" Utterances: 319 AoA: 17.5



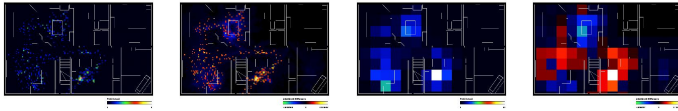
"close" Utterances: 635 AoA: 18.8



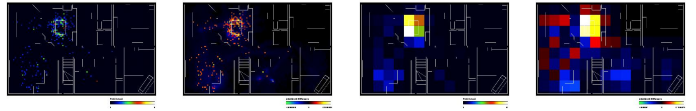
"cloth" Utterances: 85 AoA: 19.4



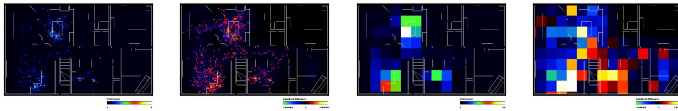
"clothes" Utterances: 366 AoA: 18.8



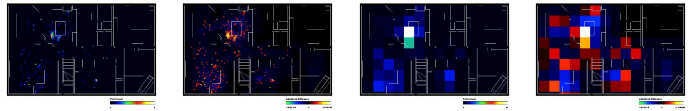
"coffee" Utterances: 181 AoA: 17.6



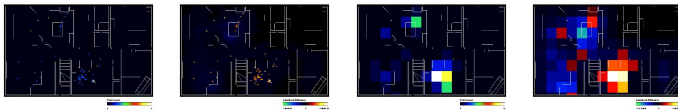
"cold" Utterances: 660 AoA: 21.3



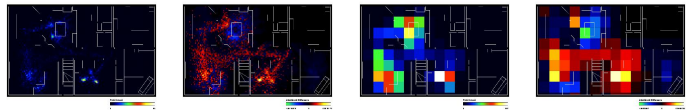
"color" Utterances: 322 AoA: 16.6



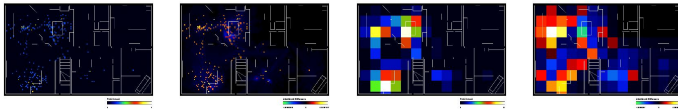
"comb" Utterances: 68 AoA: 17.8



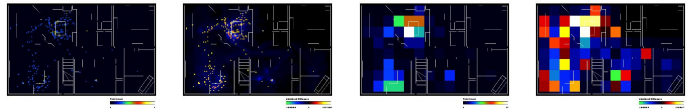
"come" Utterances: 5,455 AoA: 15.6



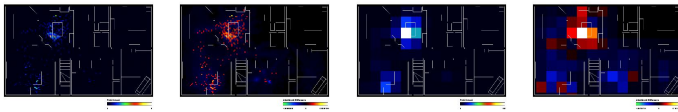
"computer" Utterances: 169 AoA: 20.7



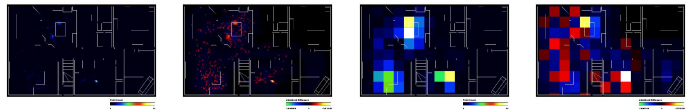
"cook" Utterances: 159 AoA: 21.4



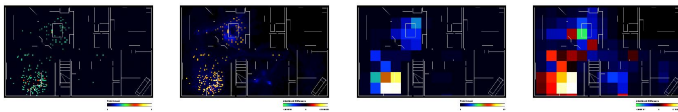
"cookie" Utterances: 311 AoA: 17.6



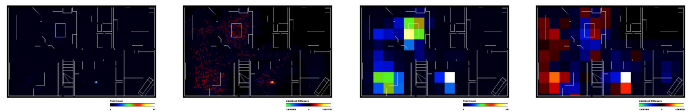
"cool" Utterances: 466 AoA: 15.7



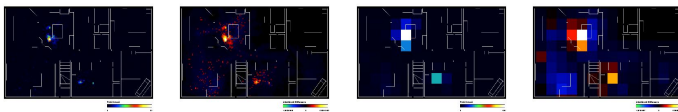
"couch" Utterances: 150 AoA: 21.1



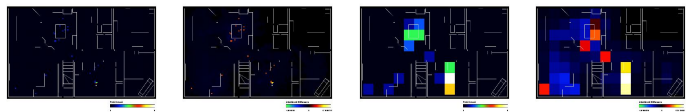
"could" Utterances: 985 AoA: 17.7



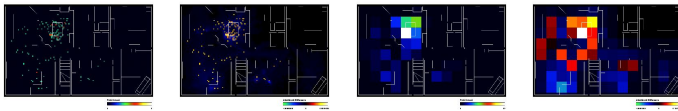
"cow" Utterances: 1,014 AoA: 16.0



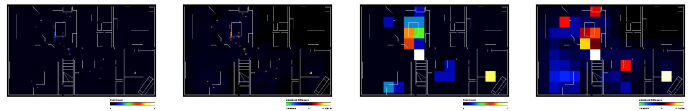
"crab" Utterances: 27 AoA: 17.6



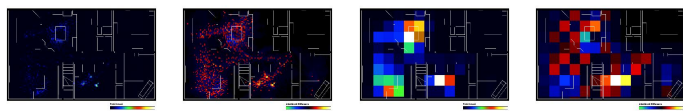
"cracker" Utterances: 99 AoA: 20.2



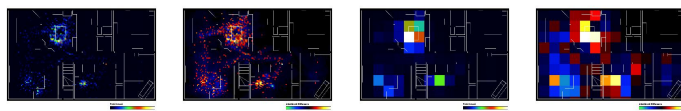
"crayon" Utterances: 40 AoA: 22.7



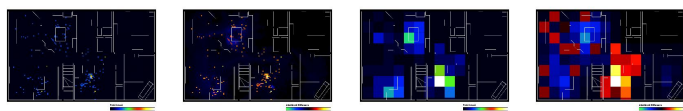
"crazy" Utterances: 1,215 AoA: 20.0



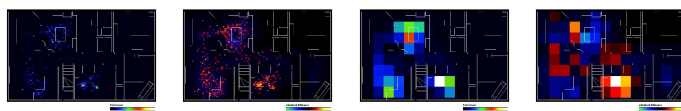
"cream" Utterances: 1,056 AoA: 20.2



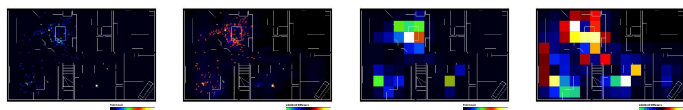
"crib" Utterances: 128 AoA: 18.4



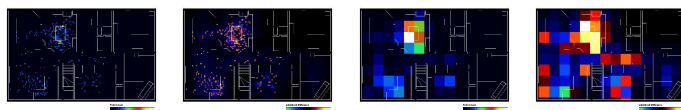
"cry" Utterances: 598 AoA: 17.1



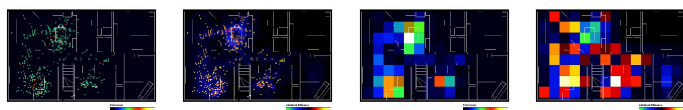
"cup" Utterances: 238 AoA: 15.6



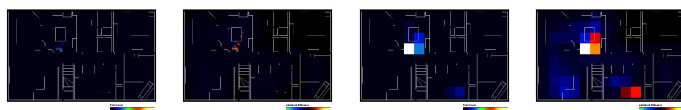
"cut" Utterances: 252 AoA: 17.5



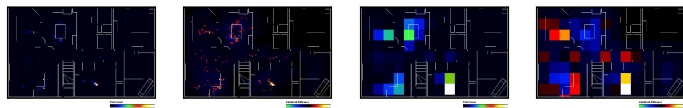
"cute" Utterances: 371 AoA: 18.7



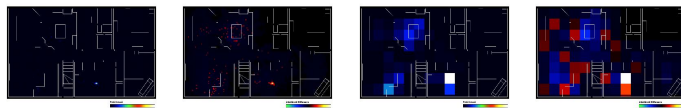
"dad" Utterances: 27 AoA: 9.5



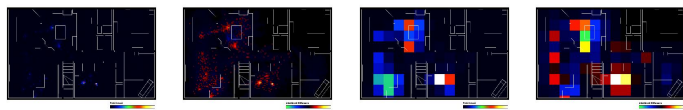
"dame" Utterances: 206 AoA: 21.1



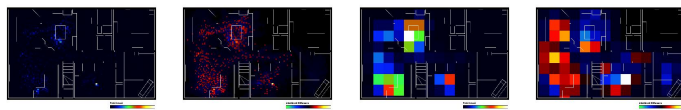
"dark" Utterances: 177 AoA: 18.1



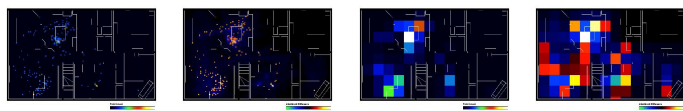
"[child name]" Utterances: 1,166 AoA: 10.6



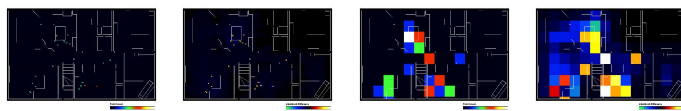
"day" Utterances: 906 AoA: 16.5



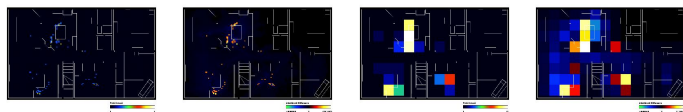
"dear" Utterances: 154 AoA: 21.1



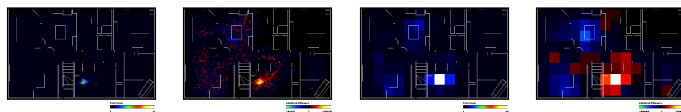
"deer" Utterances: 23 AoA: 18.7



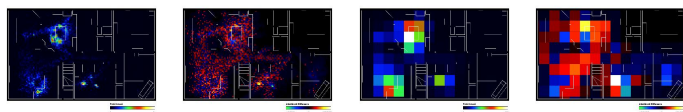
"diamond" Utterances: 88 AoA: 19.8



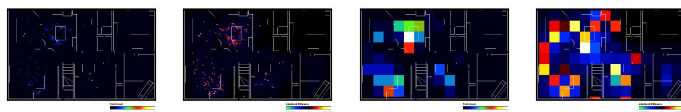
"diaper" Utterances: 1,044 AoA: 17.5



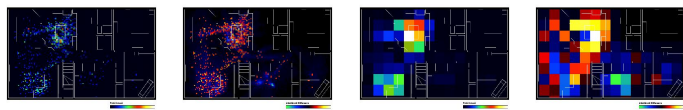
"did" Utterances: 5,945 AoA: 19.9



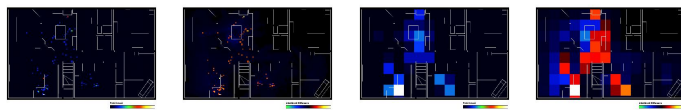
"ding" Utterances: 127 AoA: 20.9



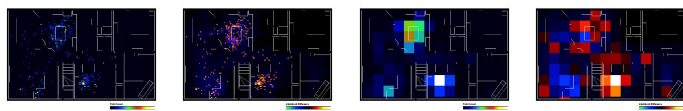
"dinner" Utterances: 754 AoA: 21.4



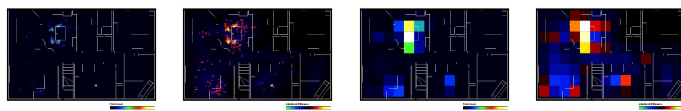
"dinosaur" Utterances: 82 AoA: 19.5



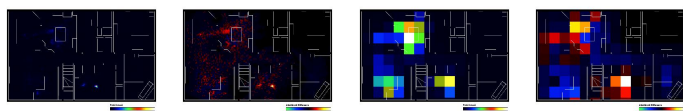
"dirty" Utterances: 253 AoA: 18.7



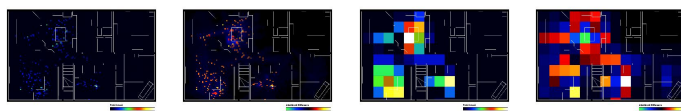
"dish" Utterances: 316 AoA: 20.1



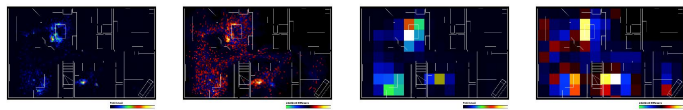
"do" Utterances: 4,122 AoA: 13.8



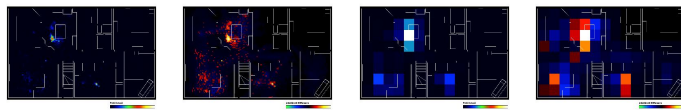
"doctor" Utterances: 168 AoA: 19.4



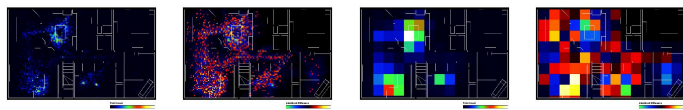
"does" Utterances: 3,743 AoA: 23.8



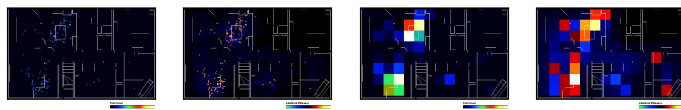
"dog" Utterances: 1,405 AoA: 16.1



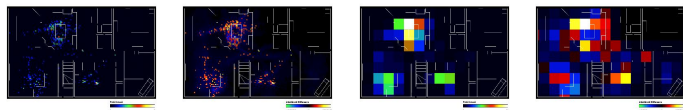
"doing" Utterances: 2,965 AoA: 20.4



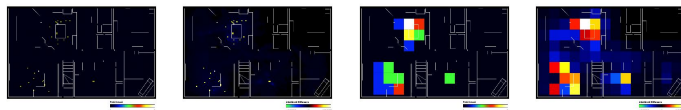
"dolphin" Utterances: 133 AoA: 21.4



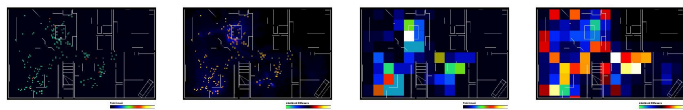
"done" Utterances: 327 AoA: 11.6



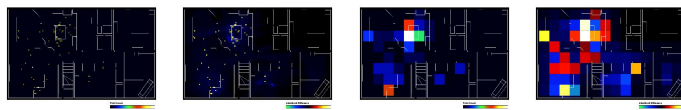
"donkey" Utterances: 26 AoA: 19.4



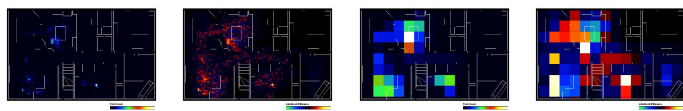
"door" Utterances: 158 AoA: 16.8



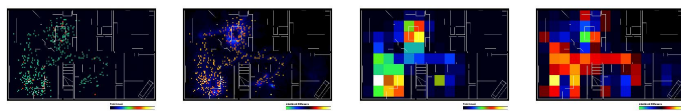
"dough" Utterances: 78 AoA: 22.3



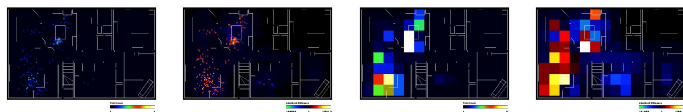
"down" Utterances: 1,350 AoA: 15.0



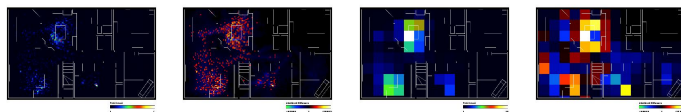
"downstairs" Utterances: 316 AoA: 19.7



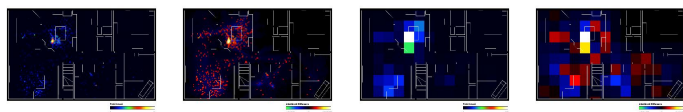
"draw" Utterances: 154 AoA: 17.4



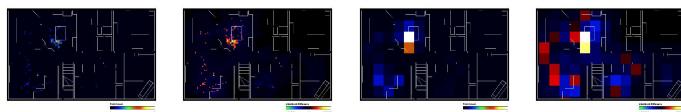
"drink" Utterances: 866 AoA: 19.8



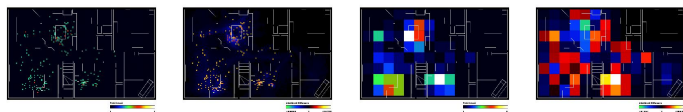
"driving" Utterances: 812 AoA: 23.4



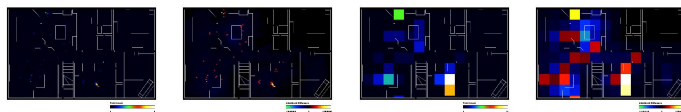
"drum" Utterances: 127 AoA: 17.5



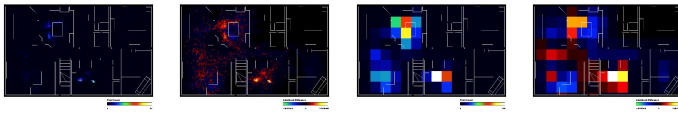
"dry" Utterances: 162 AoA: 19.3



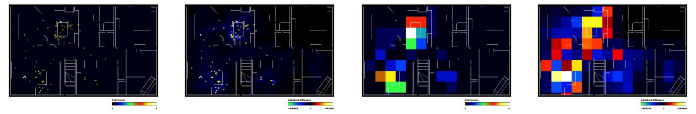
"duck" Utterances: 79 AoA: 11.3



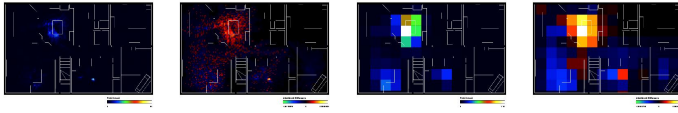
"dude" Utterances: 2,145 AoA: 16.1



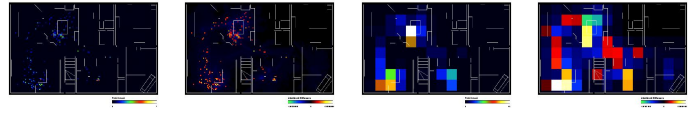
"dump" Utterances: 65 AoA: 18.0



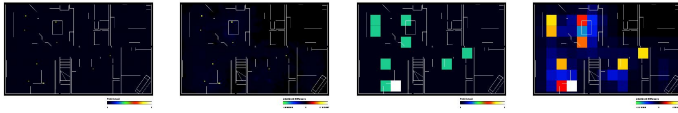
"eat" Utterances: 4,662 AoA: 19.4



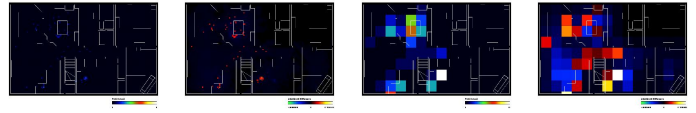
"elephant" Utterances: 221 AoA: 17.7



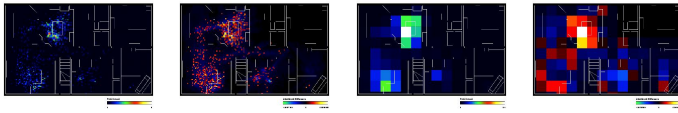
"[sister name]" Utterances: 8 AoA: 21.3



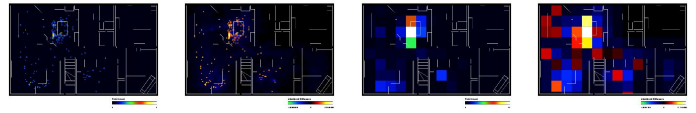
"elmo" Utterances: 71 AoA: 18.7



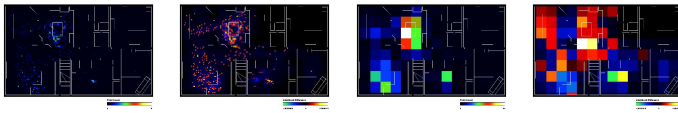
"else" Utterances: 849 AoA: 19.5



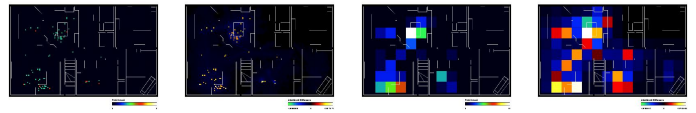
"empty" Utterances: 141 AoA: 19.4



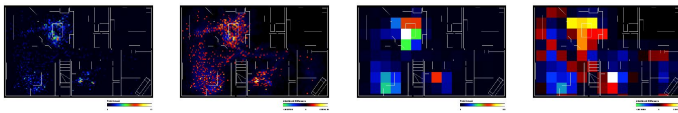
"end" Utterances: 392 AoA: 20.4



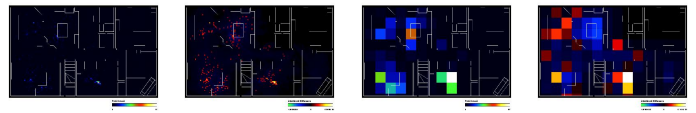
"engine" Utterances: 66 AoA: 18.8



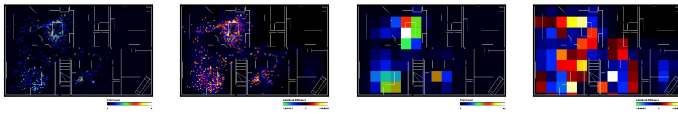
"enough" Utterances: 1,056 AoA: 21.5



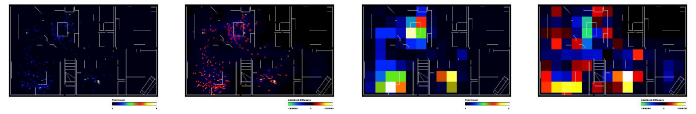
"eye" Utterances: 305 AoA: 14.6



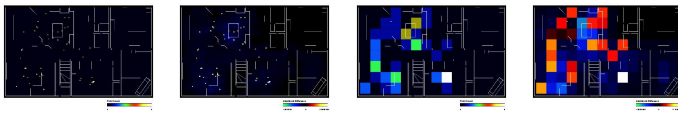
"face" Utterances: 629 AoA: 19.9



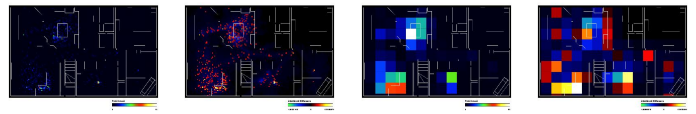
"fall" Utterances: 297 AoA: 16.4



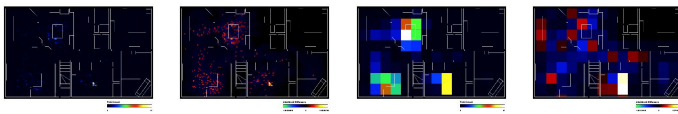
"fan" Utterances: 40 AoA: 16.6



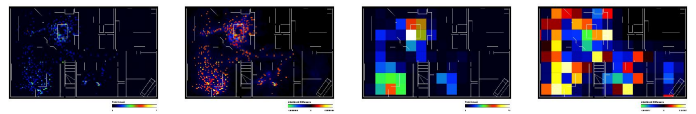
"far" Utterances: 542 AoA: 25.1



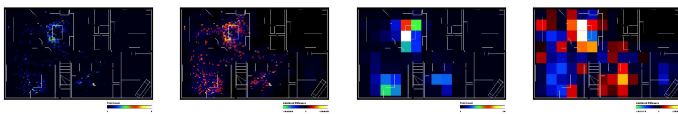
"fast" Utterances: 421 AoA: 20.0



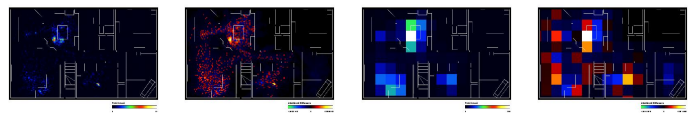
"feel" Utterances: 510 AoA: 17.6



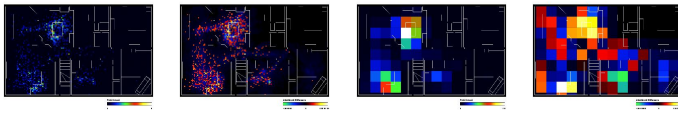
"fell" Utterances: 547 AoA: 20.4



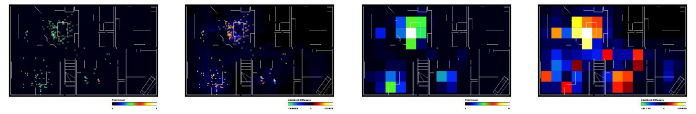
"find" Utterances: 1,321 AoA: 19.3



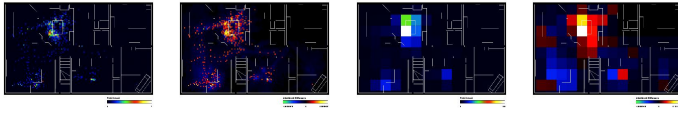
"fine" Utterances: 799 AoA: 20.9



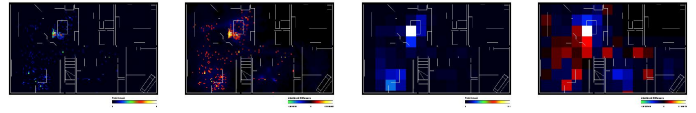
"finger" Utterances: 157 AoA: 18.8



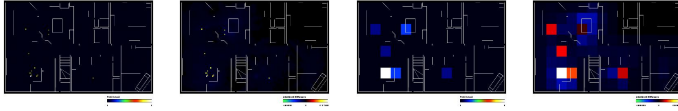
"finish" Utterances: 552 AoA: 20.2



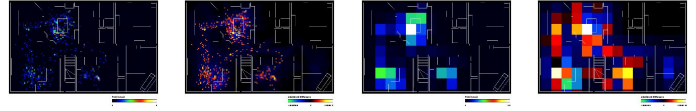
"fire" Utterances: 312 AoA: 18.0



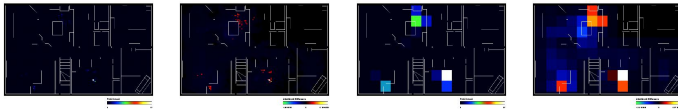
"firetruck" Utterances: 13 AoA: 18.2



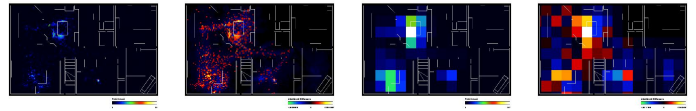
"first" Utterances: 513 AoA: 16.1



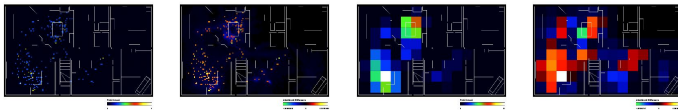
"fish" Utterances: 66 AoA: 9.6



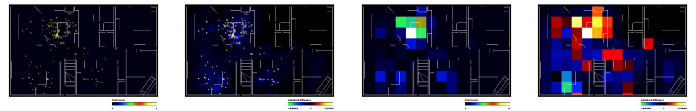
"five" Utterances: 1,688 AoA: 17.6



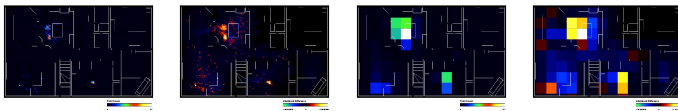
"fix" Utterances: 156 AoA: 21.7



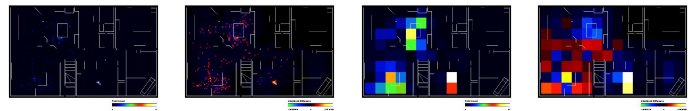
"floor" Utterances: 113 AoA: 15.0



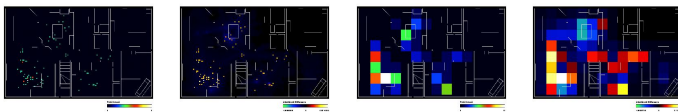
"flower" Utterances: 682 AoA: 16.1



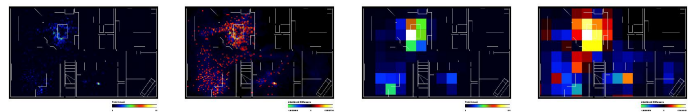
"fly" Utterances: 252 AoA: 18.0



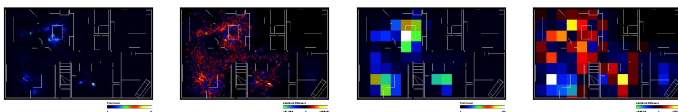
"fold" Utterances: 64 AoA: 21.3



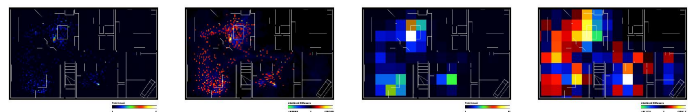
"food" Utterances: 788 AoA: 20.0



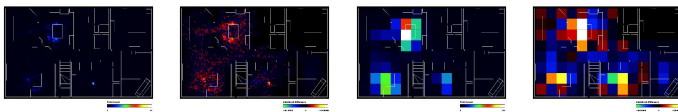
"for" Utterances: 3,757 AoA: 15.4



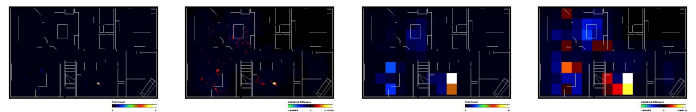
"found" Utterances: 505 AoA: 20.4



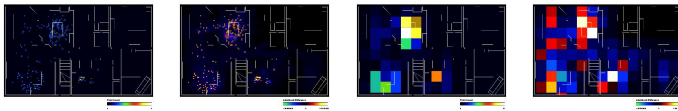
"four" Utterances: 1,753 AoA: 18.8



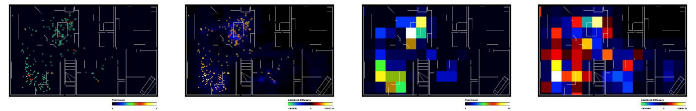
"fox" Utterances: 179 AoA: 18.5



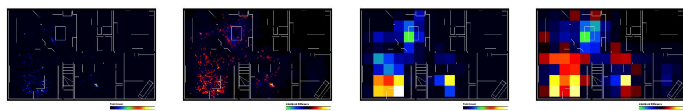
"fresh" Utterances: 165 AoA: 21.7



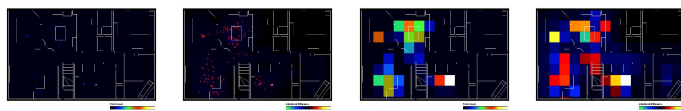
"friday" Utterances: 170 AoA: 19.7



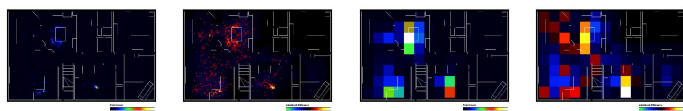
"frog" Utterances: 663 AoA: 16.6



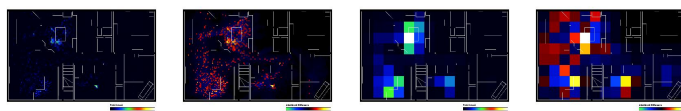
"from" Utterances: 141 AoA: 10.7



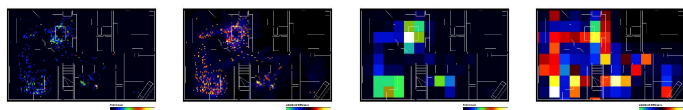
"full" Utterances: 1,103 AoA: 21.0



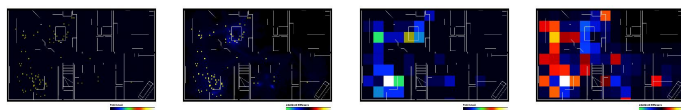
"fun" Utterances: 925 AoA: 21.4



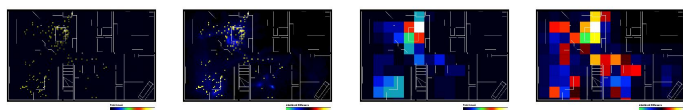
"funny" Utterances: 425 AoA: 18.0



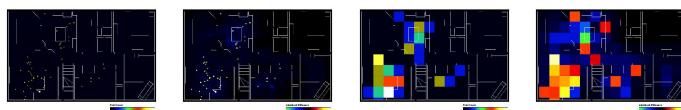
"garage" Utterances: 61 AoA: 18.8



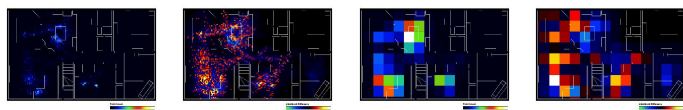
"garbage" Utterances: 107 AoA: 18.7



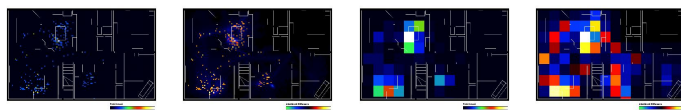
"[name 2]" Utterances: 32 AoA: 18.5



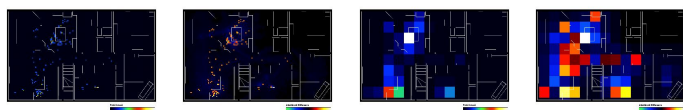
"get" Utterances: 2,705 AoA: 15.0



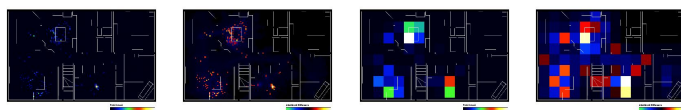
"gimme" Utterances: 184 AoA: 24.4



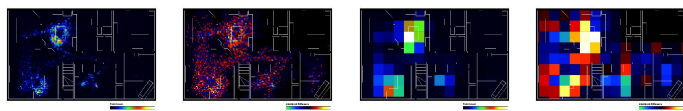
"giraffe" Utterances: 139 AoA: 18.4



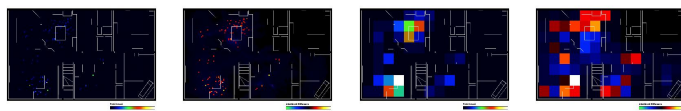
"girl" Utterances: 189 AoA: 18.7



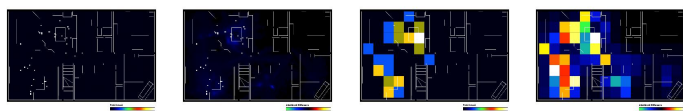
"give" Utterances: 3,863 AoA: 22.7



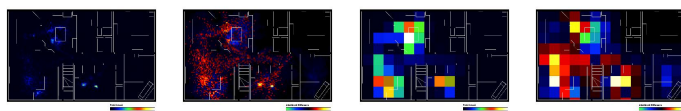
"glasses" Utterances: 87 AoA: 17.1



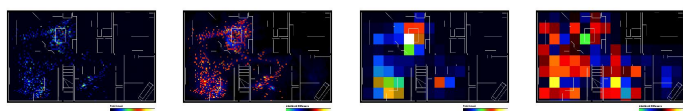
"glider" Utterances: 19 AoA: 22.5



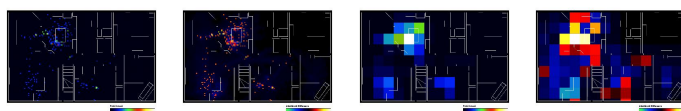
"go" Utterances: 6,104 AoA: 15.0



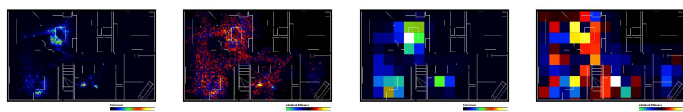
"god" Utterances: 698 AoA: 19.4



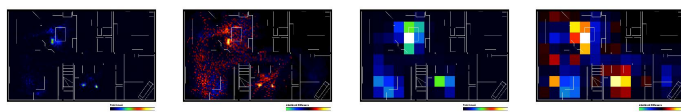
"gone" Utterances: 181 AoA: 13.7



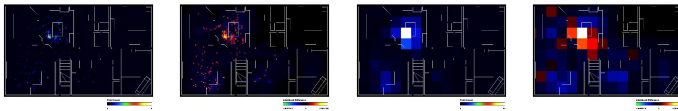
"gonna" Utterances: 6,219 AoA: 21.4



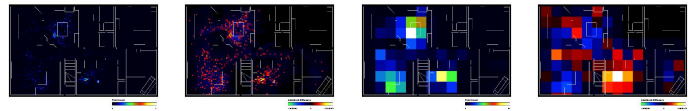
"good" Utterances: 4,638 AoA: 16.3



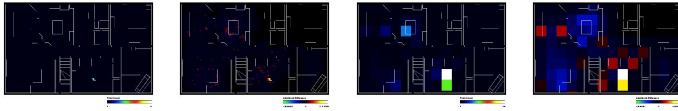
"goodbye" Utterances: 200 AoA: 17.7



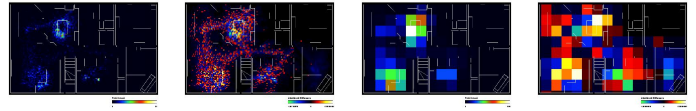
"goodness" Utterances: 676 AoA: 22.3



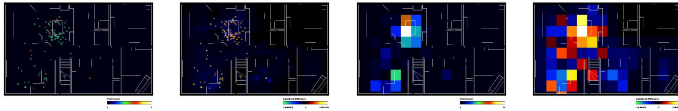
"goodnight" Utterances: 280 AoA: 21.7



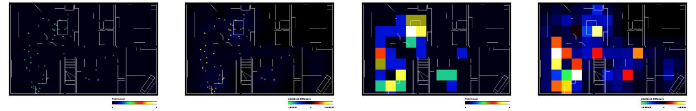
"got" Utterances: 3,178 AoA: 20.8



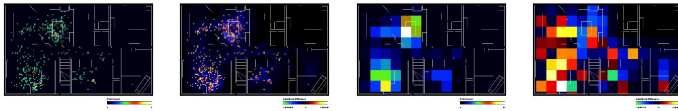
"grape" Utterances: 90 AoA: 18.5



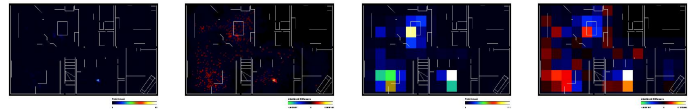
"gray" Utterances: 41 AoA: 17.6



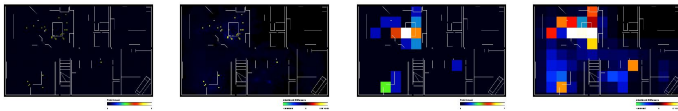
"great" Utterances: 357 AoA: 20.1



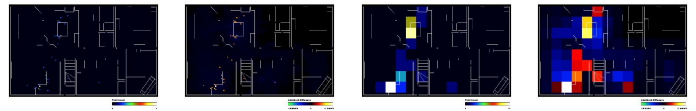
"green" Utterances: 743 AoA: 17.6



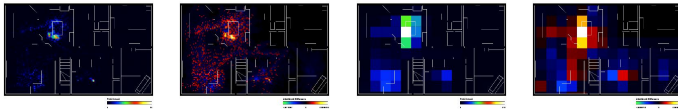
"guava" Utterances: 21 AoA: 18.3



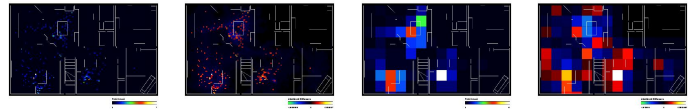
"gum" Utterances: 32 AoA: 17.7



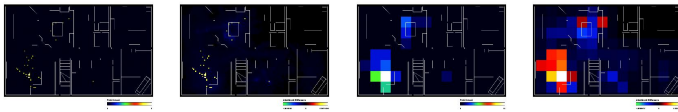
"had" Utterances: 2,449 AoA: 17.8



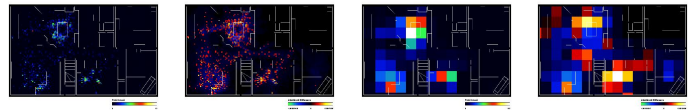
"hair" Utterances: 265 AoA: 17.5



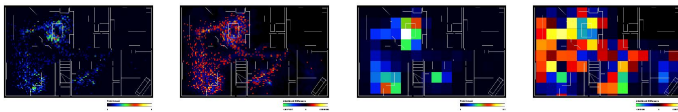
"hammer" Utterances: 32 AoA: 21.8



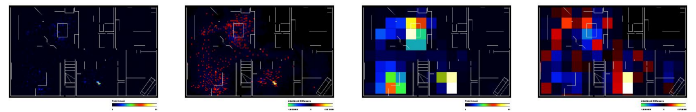
"hand" Utterances: 899 AoA: 17.9



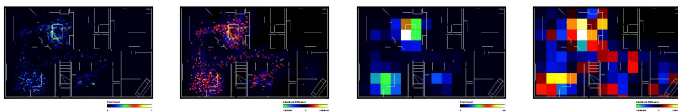
"happened" Utterances: 1,175 AoA: 21.2



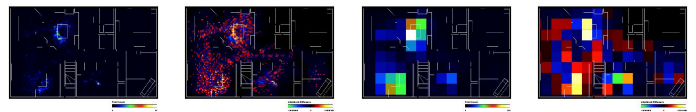
"happy" Utterances: 582 AoA: 18.1



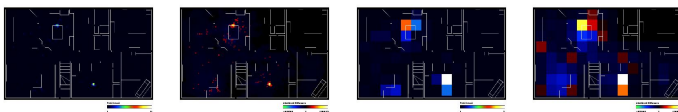
"hard" Utterances: 679 AoA: 20.8



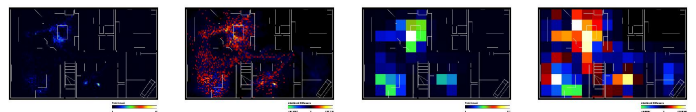
"has" Utterances: 2,291 AoA: 24.4



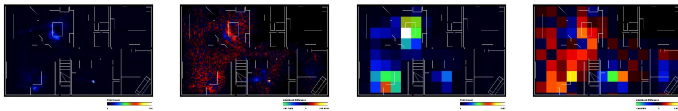
"hat" Utterances: 239 AoA: 16.9



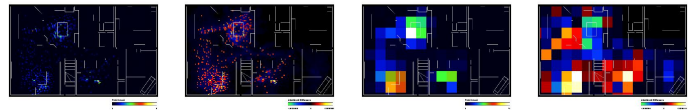
"have" Utterances: 3,181 AoA: 15.0



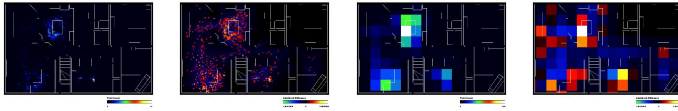
"he" Utterances: 17,604 AoA: 22.4



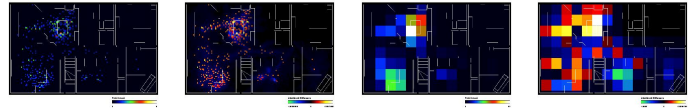
"head" Utterances: 553 AoA: 19.5



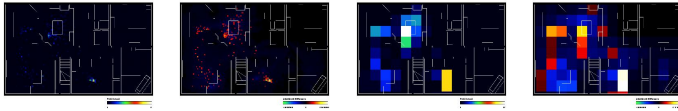
"hear" Utterances: 1,133 AoA: 24.1



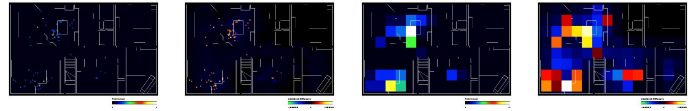
"heard" Utterances: 373 AoA: 23.4



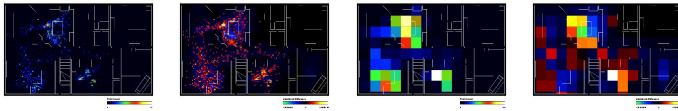
"heart" Utterances: 208 AoA: 16.8



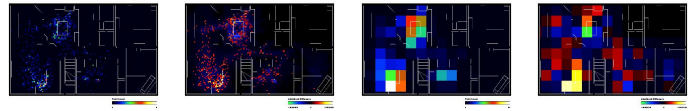
"helicopter" Utterances: 66 AoA: 18.0



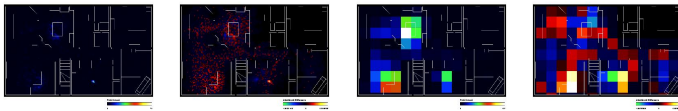
"hello" Utterances: 1,177 AoA: 16.8



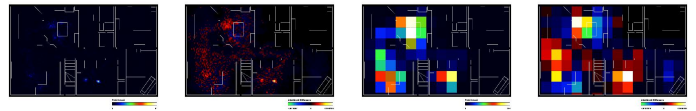
"help" Utterances: 641 AoA: 17.7



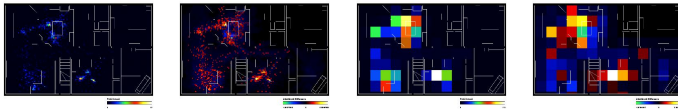
"her" Utterances: 2,590 AoA: 22.4



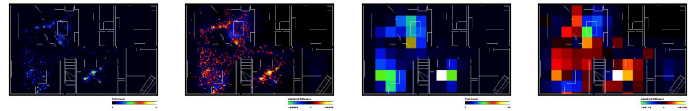
"here" Utterances: 3,422 AoA: 13.6



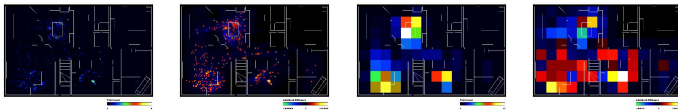
"hey" Utterances: 1,114 AoA: 11.7



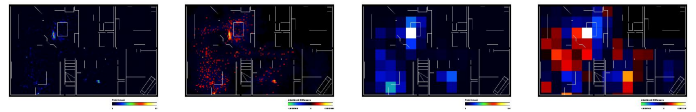
"hi" Utterances: 917 AoA: 12.7



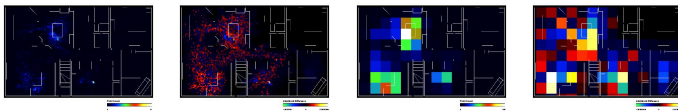
"hide" Utterances: 371 AoA: 20.3



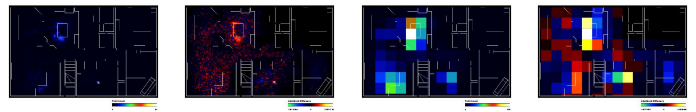
"high" Utterances: 693 AoA: 17.7



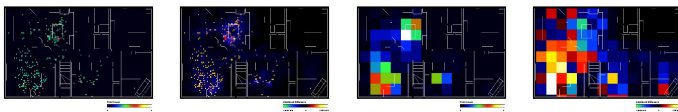
"him" Utterances: 2,842 AoA: 16.5



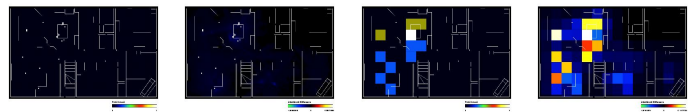
"his" Utterances: 4,728 AoA: 20.5



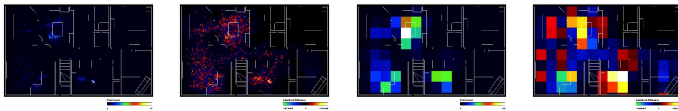
"hit" Utterances: 226 AoA: 20.3



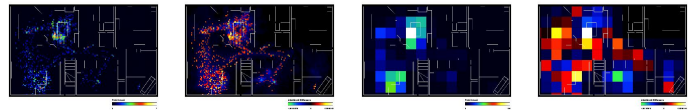
"hockey" Utterances: 9 AoA: 19.3



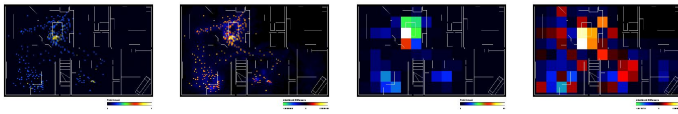
"hold" Utterances: 1,165 AoA: 18.5



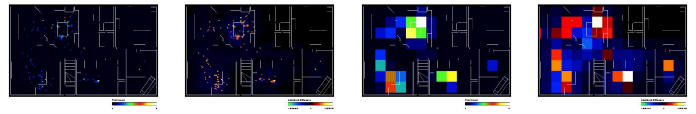
"home" Utterances: 906 AoA: 19.9



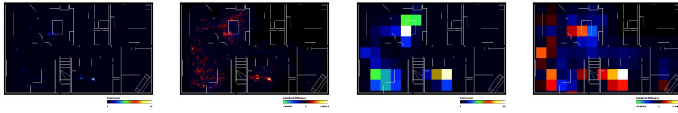
"honey" Utterances: 287 AoA: 19.7



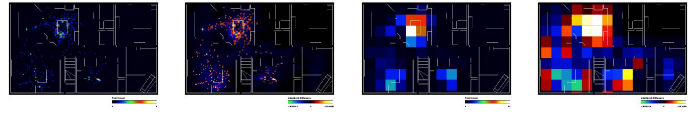
"hop" Utterances: 130 AoA: 25.2



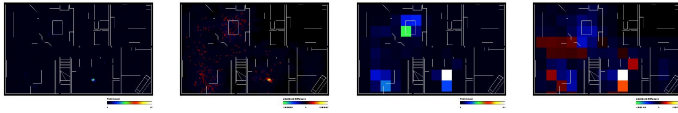
"horse" Utterances: 849 AoA: 18.8



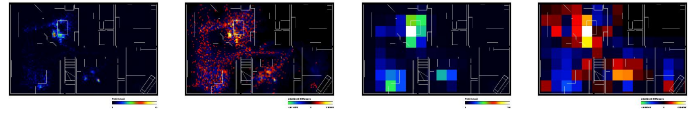
"hot" Utterances: 321 AoA: 16.8



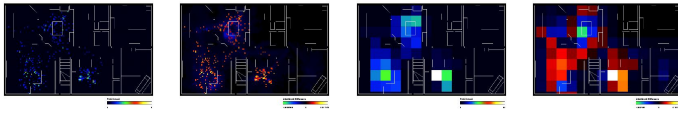
"house" Utterances: 452 AoA: 15.7



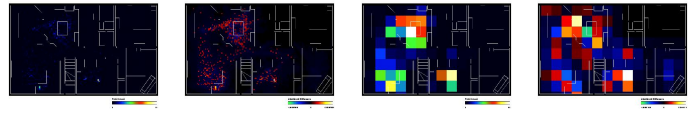
"how" Utterances: 3,999 AoA: 17.8



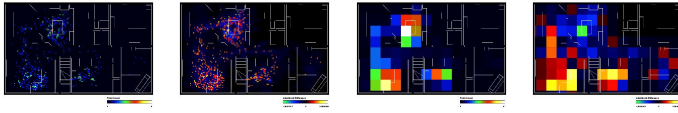
"hug" Utterances: 279 AoA: 19.9



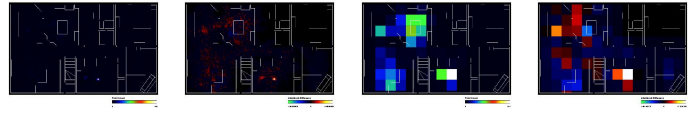
"hungry" Utterances: 576 AoA: 16.3



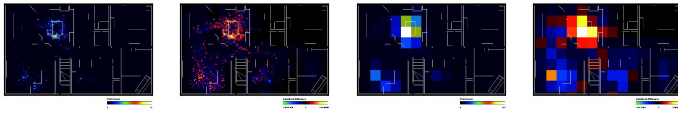
"hurt" Utterances: 611 AoA: 22.4



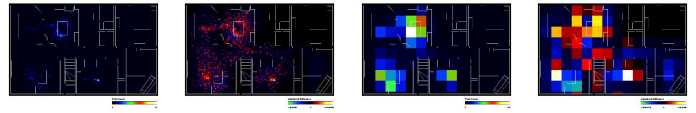
"i" Utterances: 2,683 AoA: 11.0



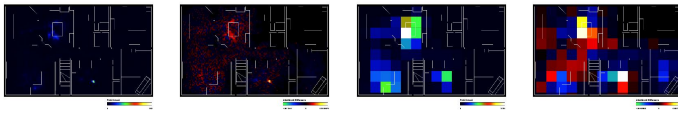
"ice" Utterances: 764 AoA: 18.1



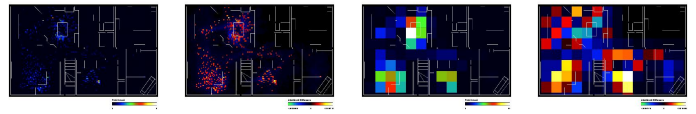
"if" Utterances: 1,365 AoA: 15.0



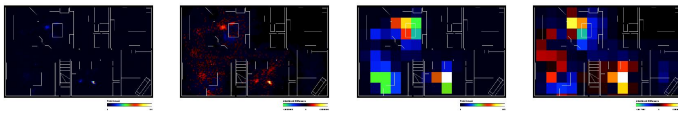
"in" Utterances: 14,265 AoA: 19.5



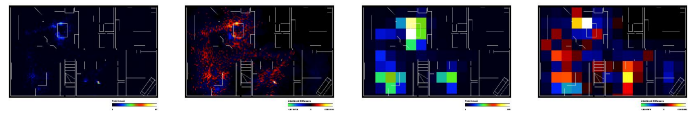
"inside" Utterances: 450 AoA: 20.0



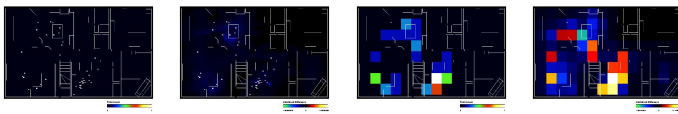
"is" Utterances: 5,495 AoA: 13.2



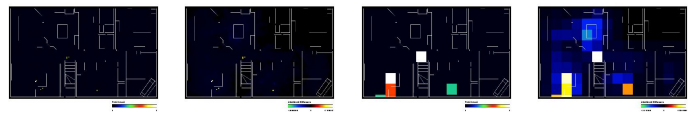
"it" Utterances: 3,975 AoA: 11.7



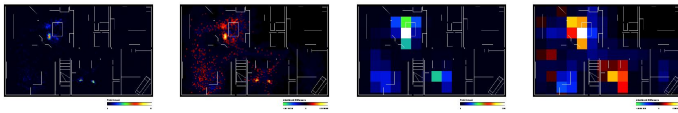
"jeans" Utterances: 40 AoA: 21.3



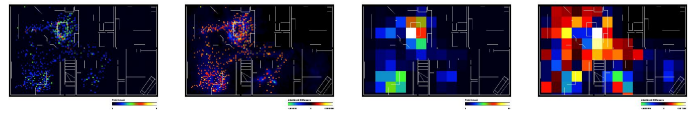
"jeep" Utterances: 18 AoA: 25.2



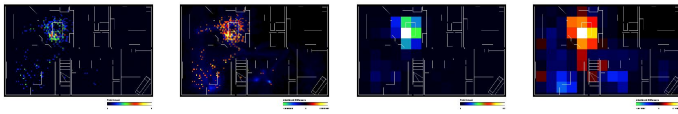
"job" Utterances: 1,497 AoA: 17.4



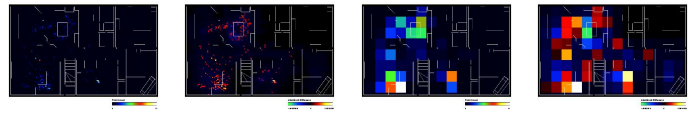
"joy" Utterances: 566 AoA: 17.1



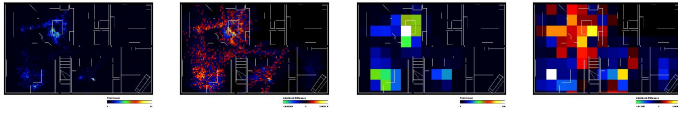
"juice" Utterances: 347 AoA: 16.7



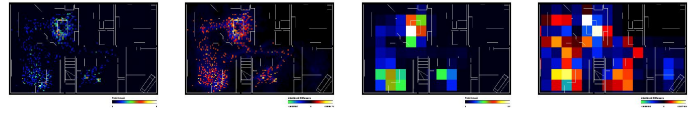
"jump" Utterances: 243 AoA: 18.9



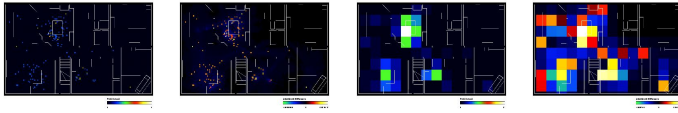
"just" Utterances: 3,344 AoA: 15.3



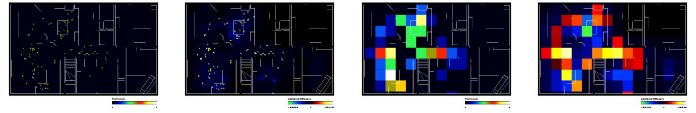
"keep" Utterances: 686 AoA: 19.4



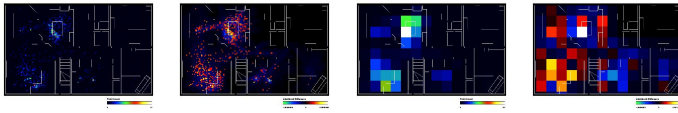
"key" Utterances: 116 AoA: 17.7



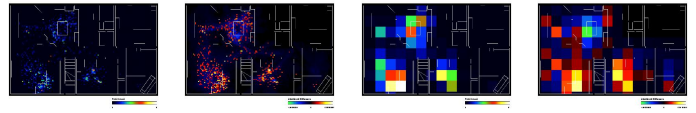
"kick" Utterances: 58 AoA: 16.1



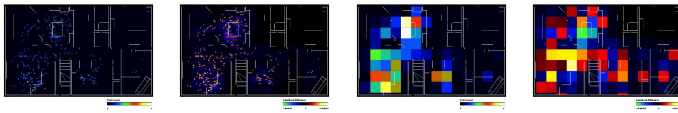
"kid" Utterances: 610 AoA: 20.2



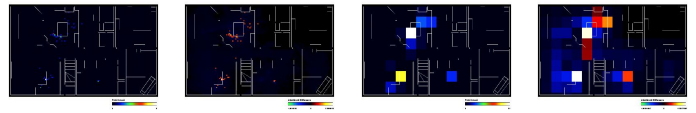
"kiss" Utterances: 815 AoA: 19.4



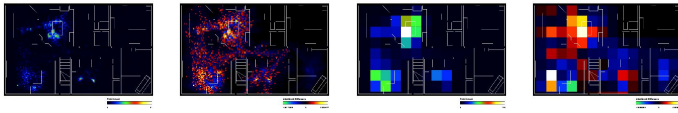
"kitchen" Utterances: 204 AoA: 20.3



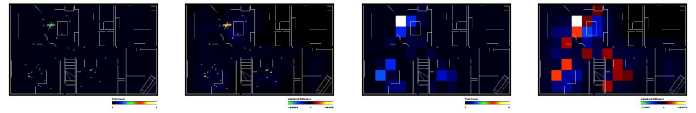
"kite" Utterances: 50 AoA: 20.9



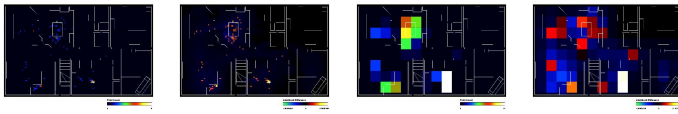
"know" Utterances: 3,891 AoA: 15.6



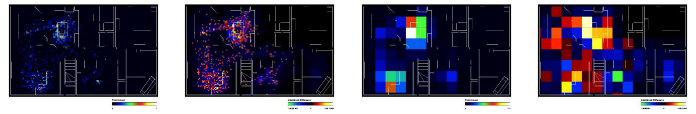
"lamp" Utterances: 55 AoA: 20.9



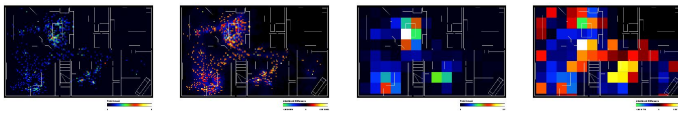
"lane" Utterances: 110 AoA: 16.2



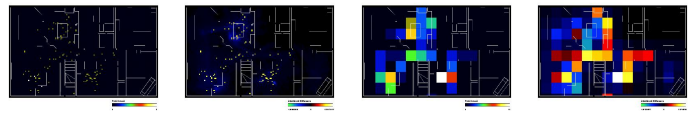
"last" Utterances: 715 AoA: 17.7



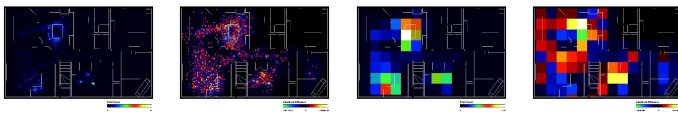
"later" Utterances: 701 AoA: 23.3



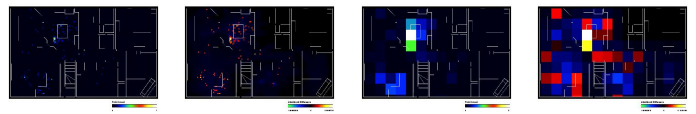
"laundry" Utterances: 76 AoA: 18.7



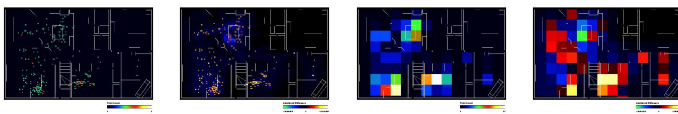
"let" Utterances: 1,715 AoA: 19.9



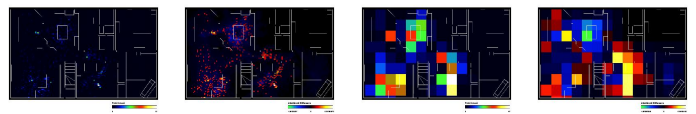
"letters" Utterances: 125 AoA: 22.4



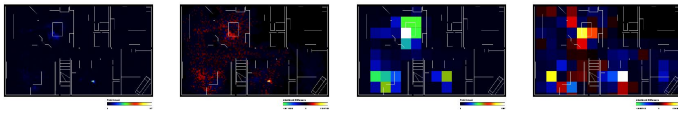
"lie" Utterances: 235 AoA: 23.5



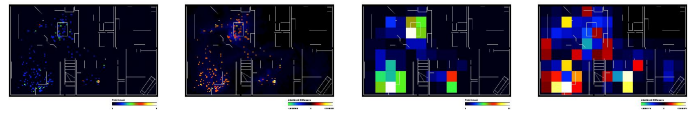
"light" Utterances: 535 AoA: 16.7



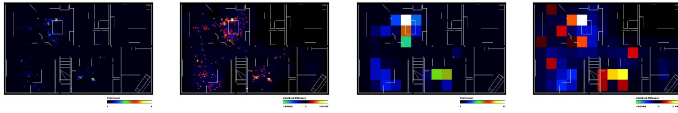
"like" Utterances: 5,929 AoA: 15.7



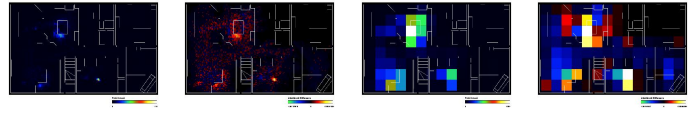
"lion" Utterances: 204 AoA: 18.3



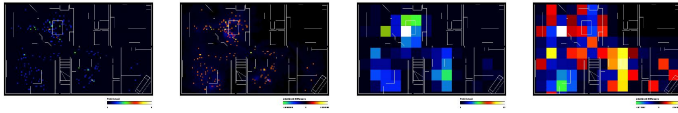
"listen" Utterances: 321 AoA: 14.5



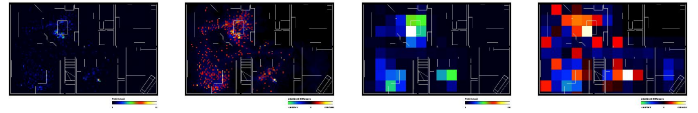
"little" Utterances: 7,280 AoA: 20.8



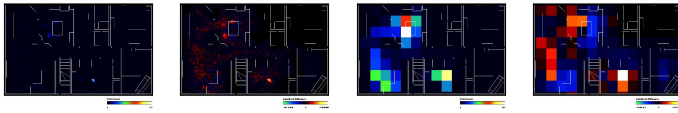
"living" Utterances: 147 AoA: 21.5



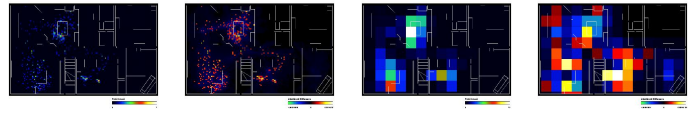
"long" Utterances: 531 AoA: 16.6



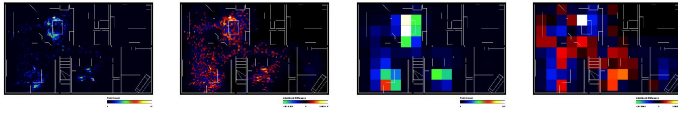
"look" Utterances: 4,254 AoA: 15.3



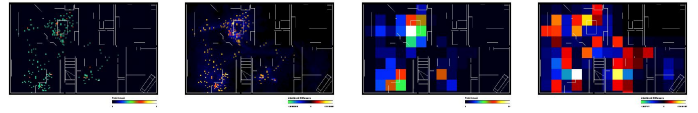
"lots" Utterances: 427 AoA: 21.0



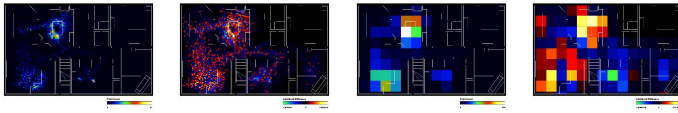
"love" Utterances: 1,742 AoA: 20.4



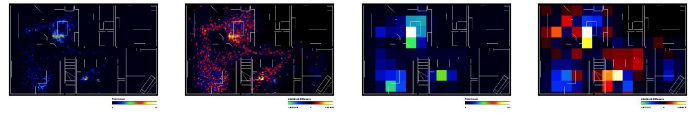
"mad" Utterances: 188 AoA: 21.2



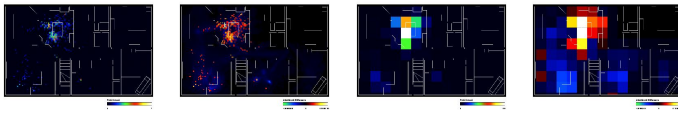
"make" Utterances: 2,872 AoA: 20.4



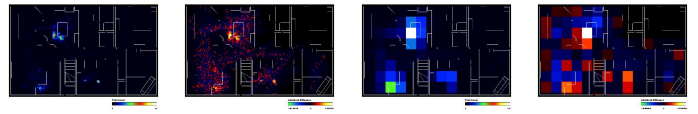
"man" Utterances: 1,960 AoA: 20.9



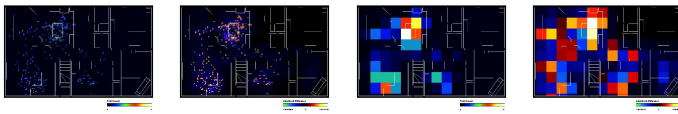
"mango" Utterances: 392 AoA: 16.5



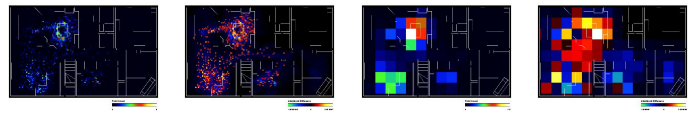
"many" Utterances: 1,956 AoA: 23.5



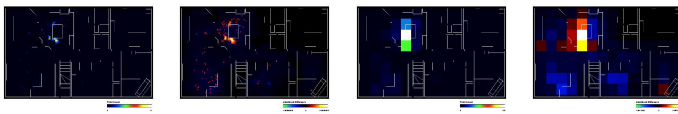
"matter" Utterances: 220 AoA: 21.2



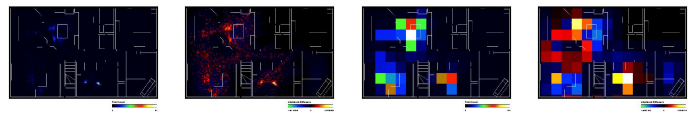
"maybe" Utterances: 841 AoA: 16.5



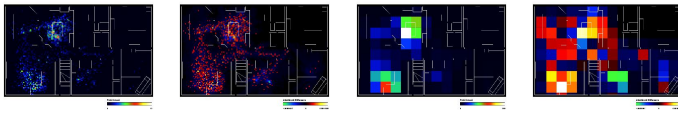
"mcdonald" Utterances: 221 AoA: 20.8



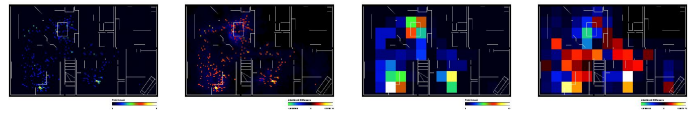
"me" Utterances: 3,546 AoA: 14.5



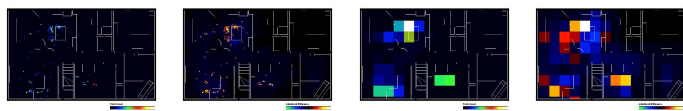
"mean" Utterances: 1,325 AoA: 20.0



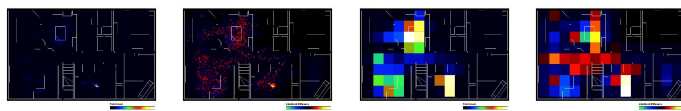
"medicine" Utterances: 259 AoA: 19.6



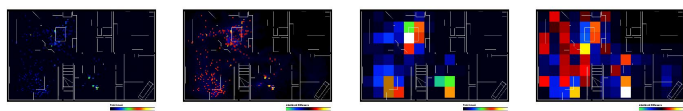
"meow" Utterances: 157 AoA: 14.4



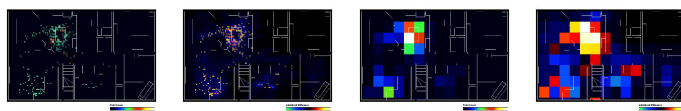
"milk" Utterances: 966 AoA: 16.5



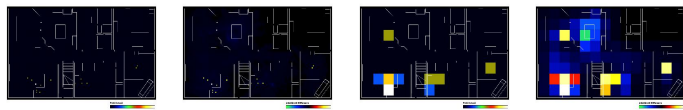
"mine" Utterances: 258 AoA: 17.1



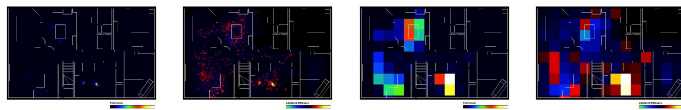
"mix" Utterances: 212 AoA: 21.0



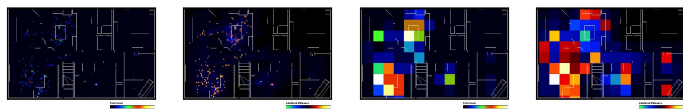
"mobile" Utterances: 16 AoA: 23.3



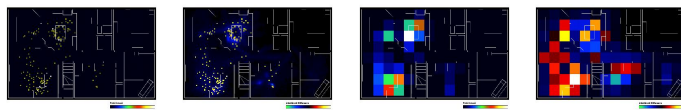
"mom" Utterances: 827 AoA: 13.1



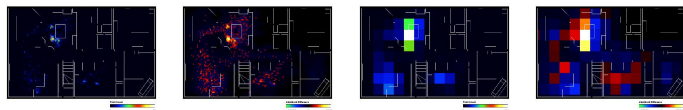
"monday" Utterances: 151 AoA: 20.2



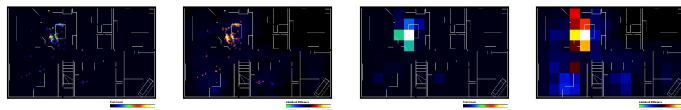
"money" Utterances: 125 AoA: 19.7



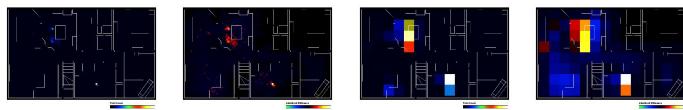
"monkey" Utterances: 1,050 AoA: 16.6



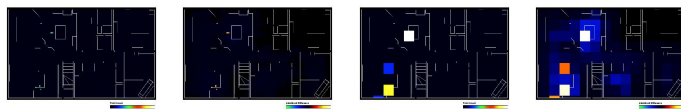
"moo" Utterances: 158 AoA: 13.5



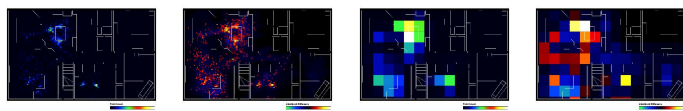
"moon" Utterances: 428 AoA: 15.2



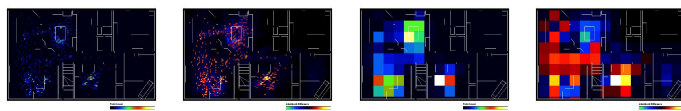
"moose" Utterances: 21 AoA: 24.3



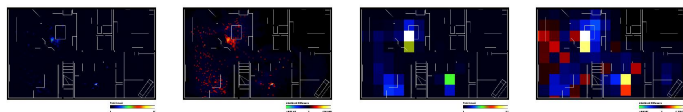
"more" Utterances: 1,231 AoA: 13.2



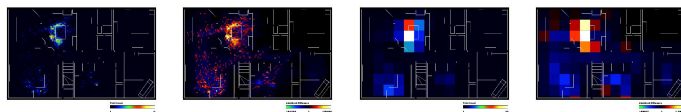
"morning" Utterances: 497 AoA: 16.1



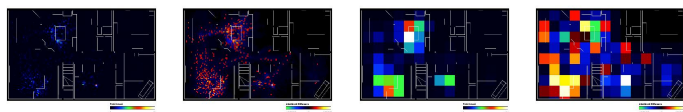
"mouse" Utterances: 740 AoA: 17.7



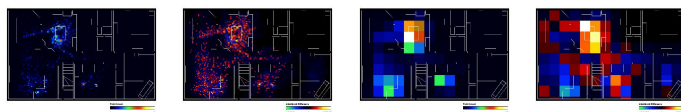
"mouth" Utterances: 1,514 AoA: 18.1



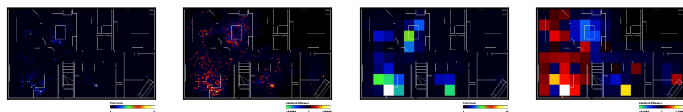
"move" Utterances: 710 AoA: 20.0



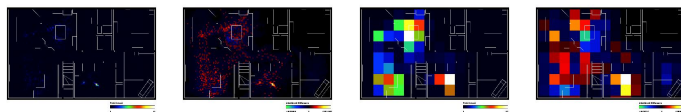
"much" Utterances: 1,268 AoA: 17.7



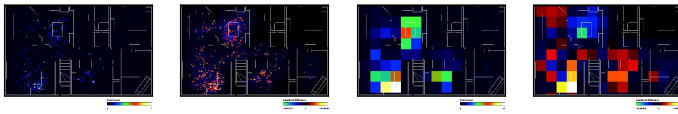
"music" Utterances: 468 AoA: 21.1



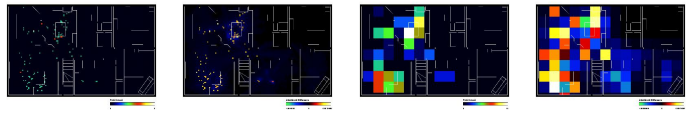
"my" Utterances: 1,275 AoA: 13.3



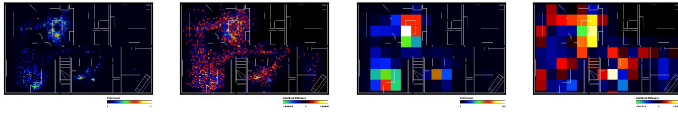
"nap" Utterances: 362 AoA: 24.1



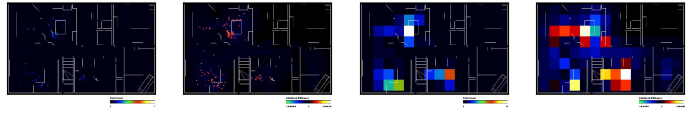
"neat" Utterances: 94 AoA: 19.9



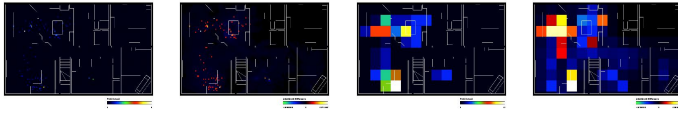
"need" Utterances: 2,230 AoA: 21.9



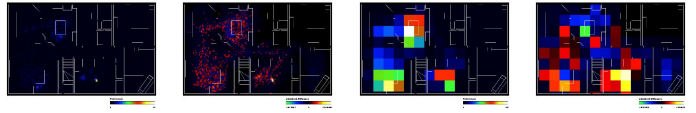
"neigh" Utterances: 121 AoA: 15.8



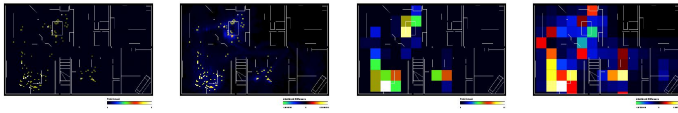
"nemo" Utterances: 95 AoA: 18.6



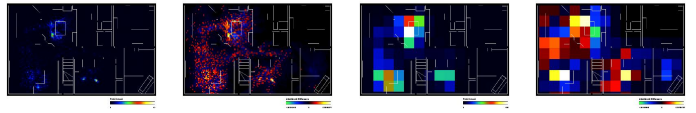
"new" Utterances: 1,101 AoA: 20.4



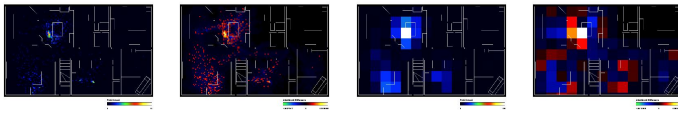
"next" Utterances: 120 AoA: 11.7



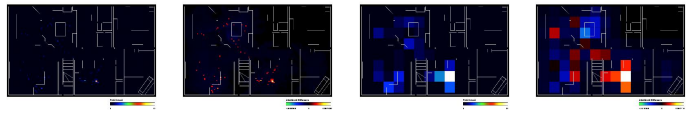
"nice" Utterances: 3,009 AoA: 19.7



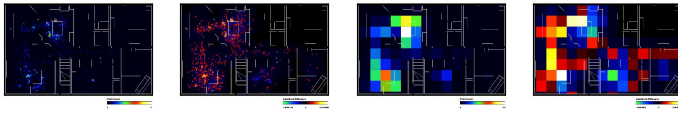
"nicely" Utterances: 636 AoA: 21.4



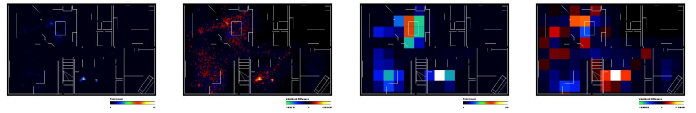
"night" Utterances: 90 AoA: 10.9



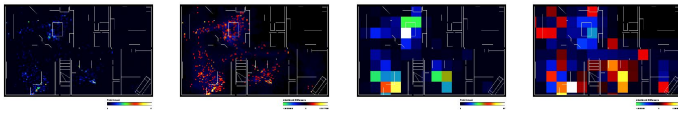
"nine" Utterances: 875 AoA: 23.8



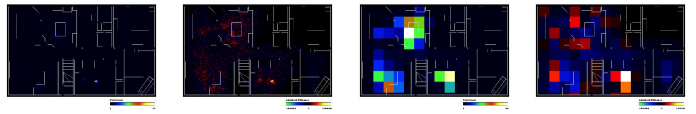
"no" Utterances: 1,671 AoA: 11.3



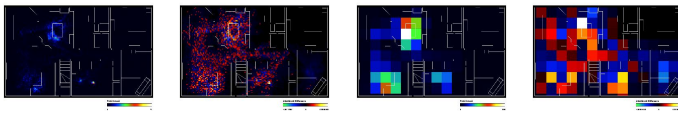
"nose" Utterances: 661 AoA: 18.1



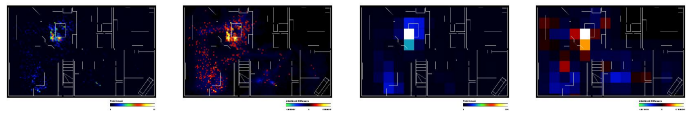
"not" Utterances: 3,467 AoA: 15.0



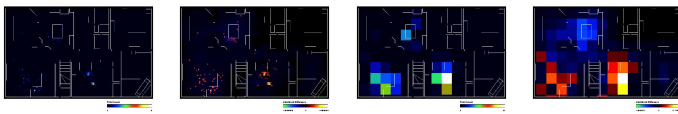
"now" Utterances: 5,746 AoA: 19.9



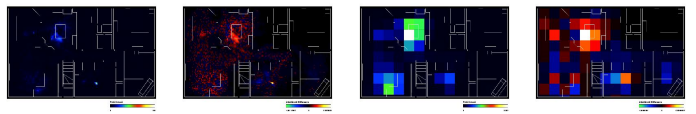
"number" Utterances: 947 AoA: 21.5



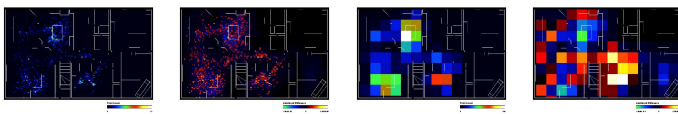
"octopus" Utterances: 124 AoA: 19.5



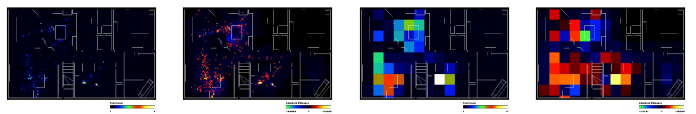
"of" Utterances: 10,327 AoA: 19.9



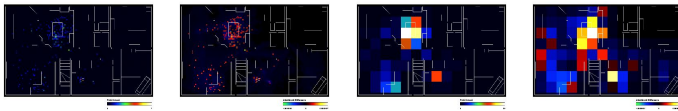
"off" Utterances: 1,007 AoA: 16.6



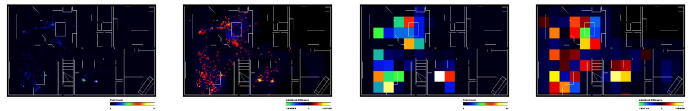
"oh" Utterances: 360 AoA: 10.0



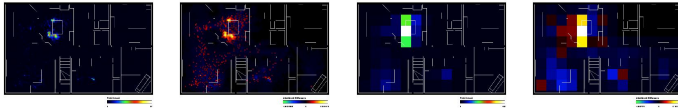
"oil" Utterances: 174 AoA: 19.4



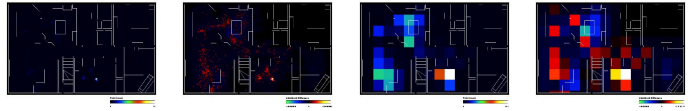
"ok" Utterances: 439 AoA: 10.0



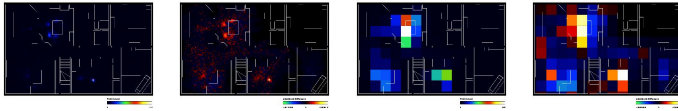
"old" Utterances: 1,215 AoA: 19.4



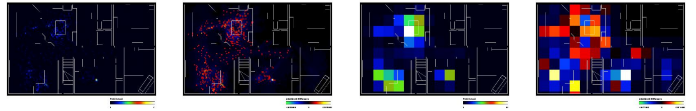
"on" Utterances: 1,130 AoA: 10.3



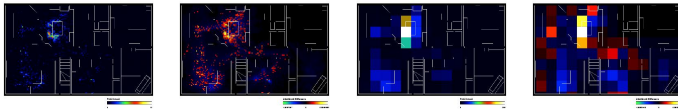
"one" Utterances: 4,684 AoA: 14.7



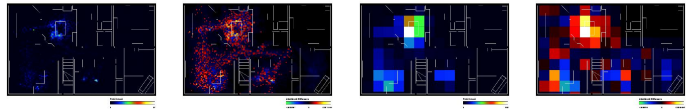
"only" Utterances: 486 AoA: 15.7



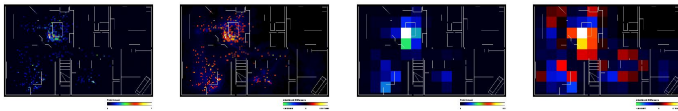
"open" Utterances: 904 AoA: 16.5



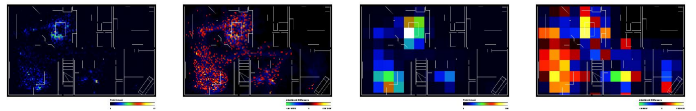
"or" Utterances: 2,551 AoA: 16.7



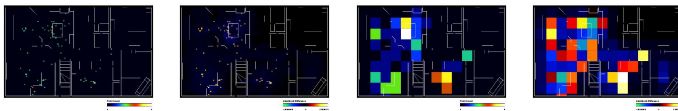
"orange" Utterances: 376 AoA: 16.9



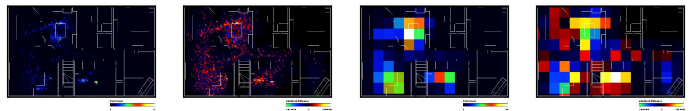
"other" Utterances: 1,490 AoA: 19.1



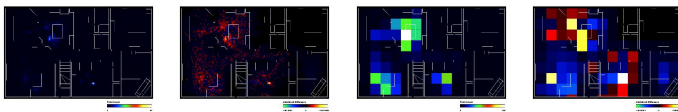
"ouch" Utterances: 71 AoA: 16.8



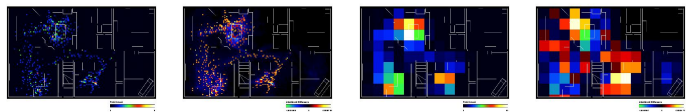
"our" Utterances: 1,490 AoA: 21.7



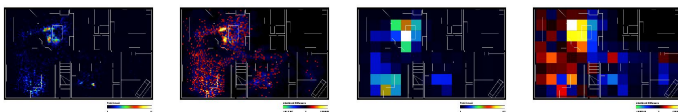
"out" Utterances: 2,012 AoA: 15.1



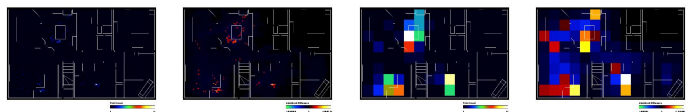
"outside" Utterances: 531 AoA: 19.3



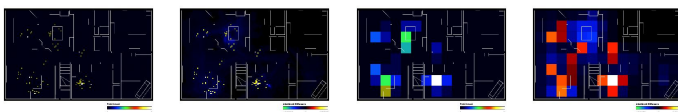
"over" Utterances: 2,367 AoA: 21.1



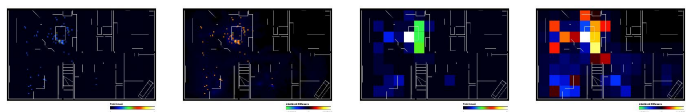
"owl" Utterances: 108 AoA: 17.7



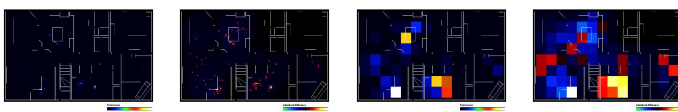
"pajamas" Utterances: 63 AoA: 19.5



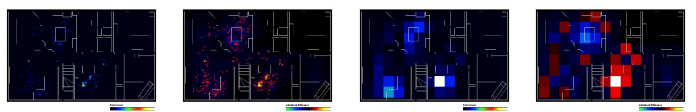
"pancakes" Utterances: 69 AoA: 22.0



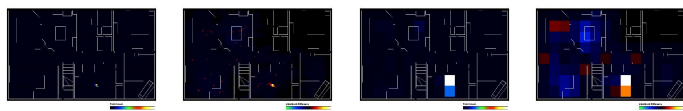
"panda" Utterances: 110 AoA: 20.3



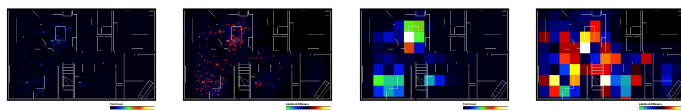
"pants" Utterances: 335 AoA: 16.9



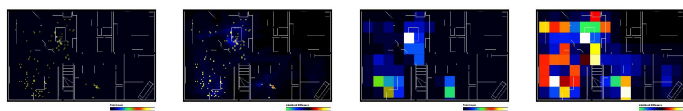
"papa" Utterances: 144 AoA: 16.9



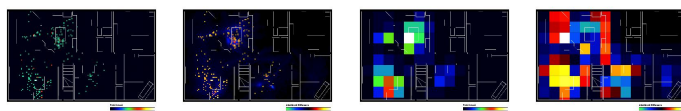
"paper" Utterances: 202 AoA: 20.7



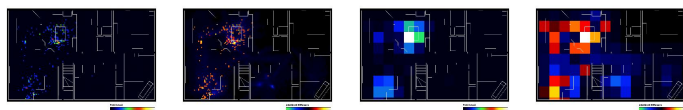
"park" Utterances: 95 AoA: 19.9



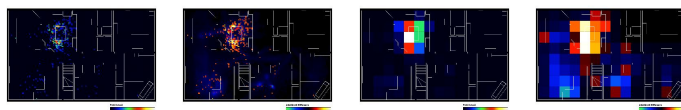
"party" Utterances: 164 AoA: 21.7



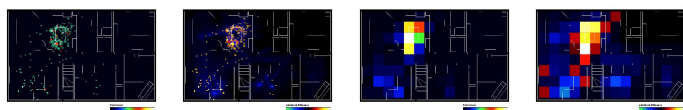
"pasta" Utterances: 197 AoA: 20.2



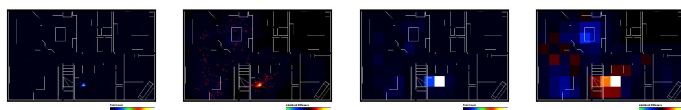
"pea" Utterances: 327 AoA: 17.9



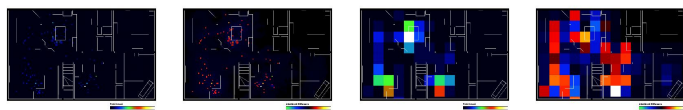
"pear" Utterances: 218 AoA: 19.4



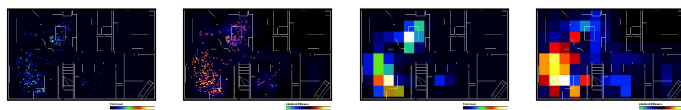
"pee" Utterances: 388 AoA: 17.5



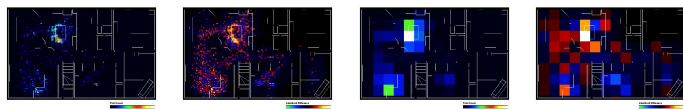
"peek" Utterances: 93 AoA: 16.2



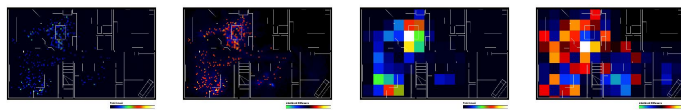
"pen" Utterances: 323 AoA: 17.0



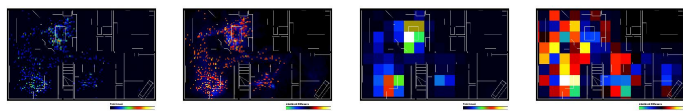
"people" Utterances: 1,321 AoA: 25.2



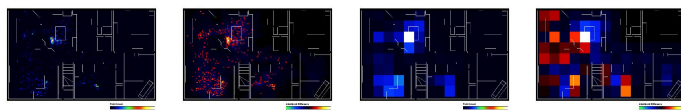
"phone" Utterances: 328 AoA: 16.6



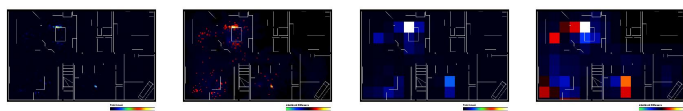
"pick" Utterances: 626 AoA: 19.3



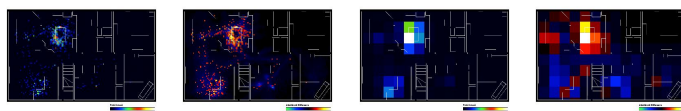
"picture" Utterances: 566 AoA: 18.4



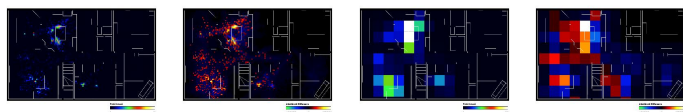
"pie" Utterances: 273 AoA: 17.9



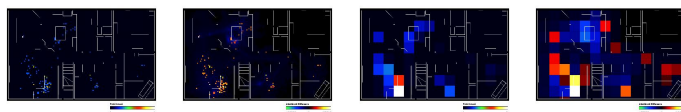
"piece" Utterances: 523 AoA: 22.3



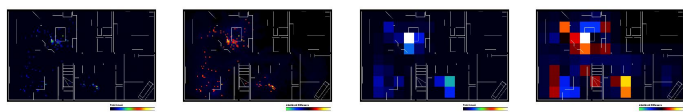
"pig" Utterances: 1,202 AoA: 18.1



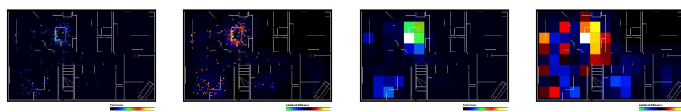
"pillow" Utterances: 118 AoA: 20.4



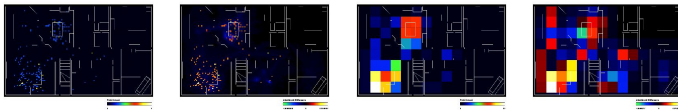
"pink" Utterances: 158 AoA: 17.5



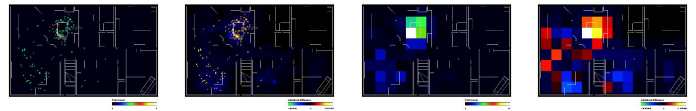
"pizza" Utterances: 140 AoA: 20.2



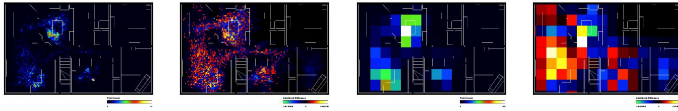
"plane" Utterances: 137 AoA: 17.5



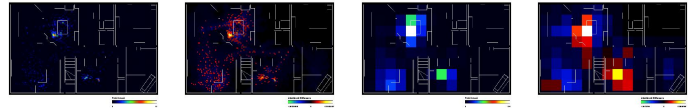
"plate" Utterances: 108 AoA: 22.0



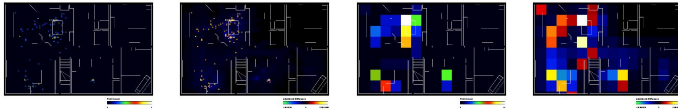
"play" Utterances: 2,712 AoA: 19.1



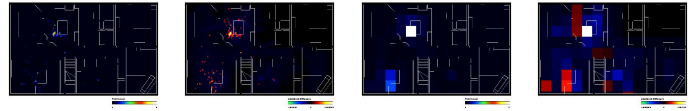
"please" Utterances: 596 AoA: 16.5



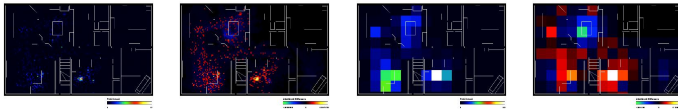
"plum" Utterances: 104 AoA: 19.5



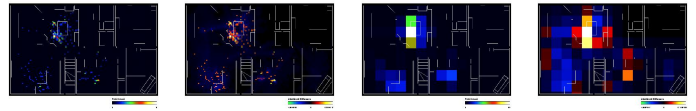
"police" Utterances: 144 AoA: 19.6



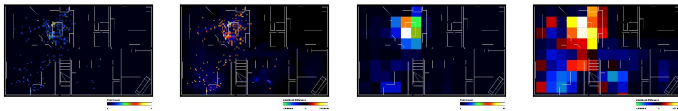
"poop" Utterances: 625 AoA: 17.5



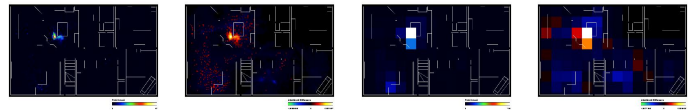
"pop" Utterances: 246 AoA: 17.6



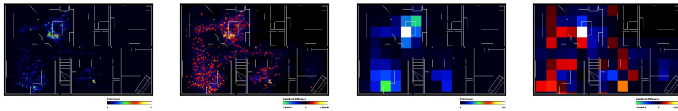
"potato" Utterances: 207 AoA: 18.7



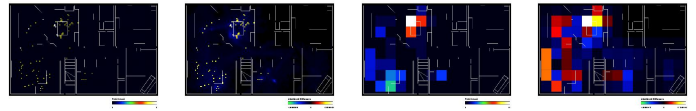
"press" Utterances: 1,446 AoA: 21.8



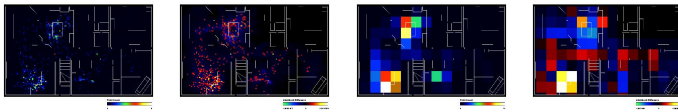
"pretty" Utterances: 1,009 AoA: 20.4



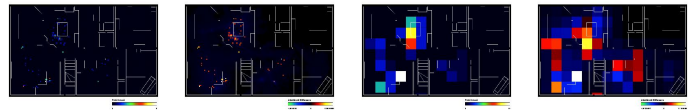
"prince" Utterances: 66 AoA: 20.8



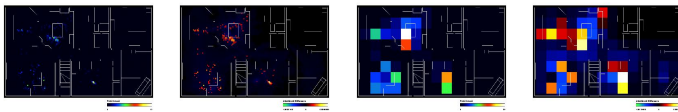
"pull" Utterances: 594 AoA: 23.3



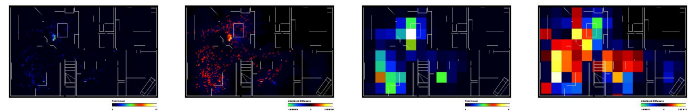
"puppy" Utterances: 91 AoA: 17.7



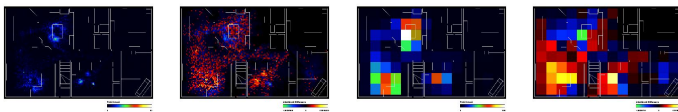
"purple" Utterances: 238 AoA: 16.8



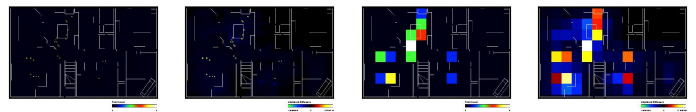
"push" Utterances: 521 AoA: 16.7



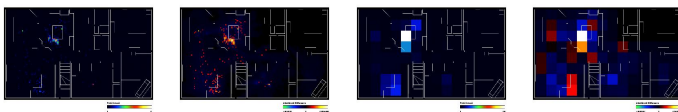
"put" Utterances: 6,320 AoA: 20.4



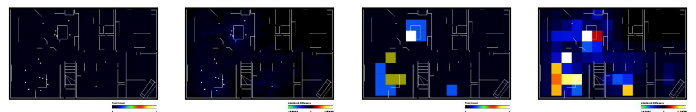
"puzzle" Utterances: 17 AoA: 15.5



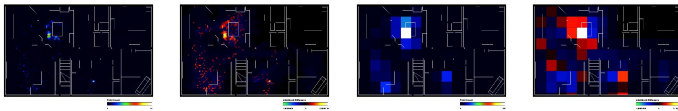
"race" Utterances: 225 AoA: 20.4



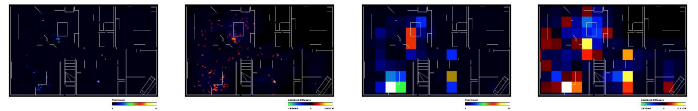
"racecar" Utterances: 13 AoA: 23.9



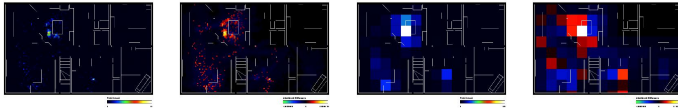
"rain" Utterances: 571 AoA: 19.4



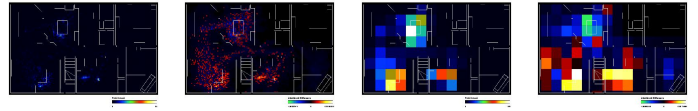
"rainbow" Utterances: 233 AoA: 23.9



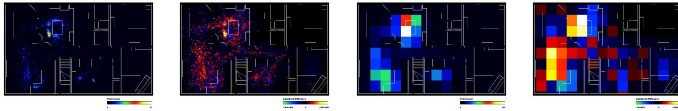
"raining" Utterances: 571 AoA: 19.4



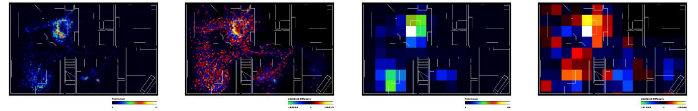
"read" Utterances: 1,447 AoA: 22.1



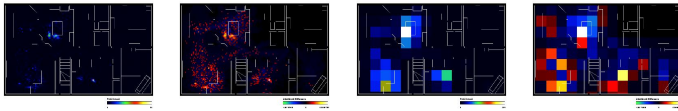
"ready" Utterances: 1,768 AoA: 18.9



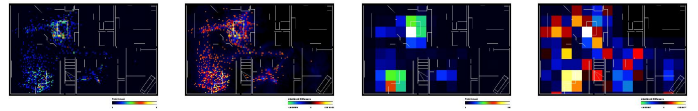
"really" Utterances: 2,881 AoA: 21.2



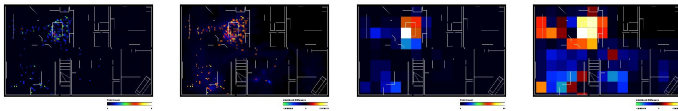
"red" Utterances: 1,369 AoA: 18.4



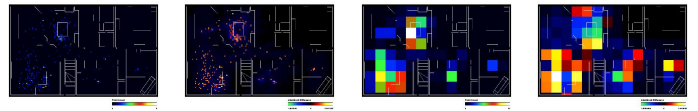
"remember" Utterances: 855 AoA: 20.9



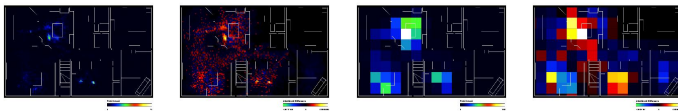
"rice" Utterances: 185 AoA: 19.9



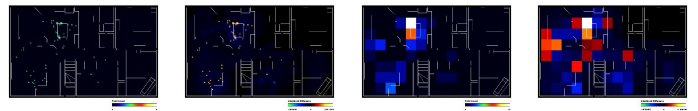
"ride" Utterances: 244 AoA: 25.2



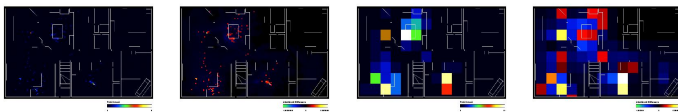
"right" Utterances: 4,460 AoA: 16.2



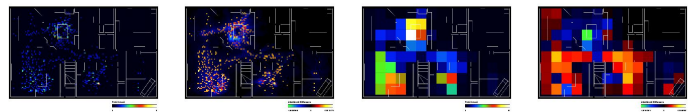
"robot" Utterances: 55 AoA: 20.2



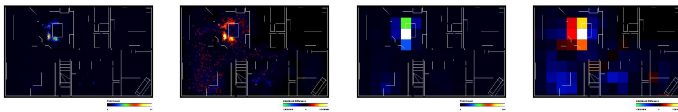
"rock" Utterances: 112 AoA: 17.7



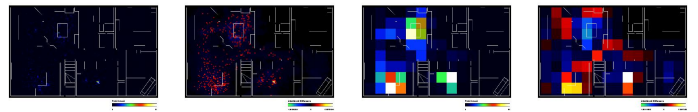
"room" Utterances: 509 AoA: 20.2



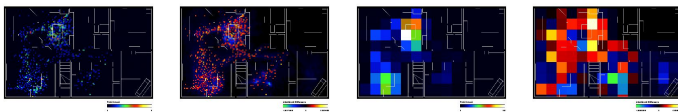
"round" Utterances: 1,599 AoA: 20.9



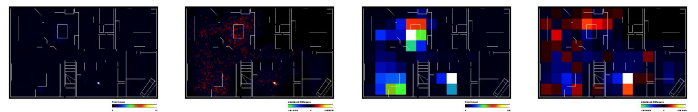
"run" Utterances: 460 AoA: 18.7



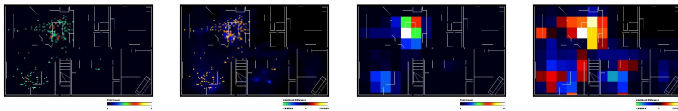
"[mother name]" Utterances: 654 AoA: 17.5



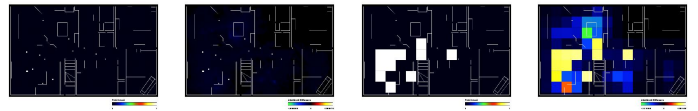
"said" Utterances: 1,461 AoA: 16.6



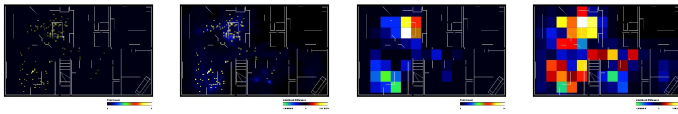
"salad" Utterances: 143 AoA: 23.3



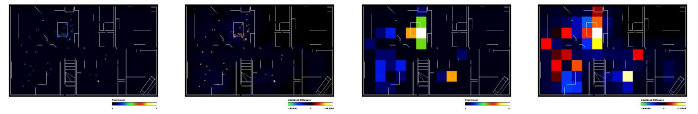
"sandals" Utterances: 6 AoA: 19.9



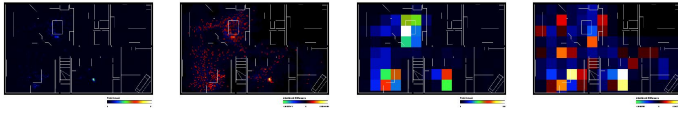
"sandwich" Utterances: 100 AoA: 23.8



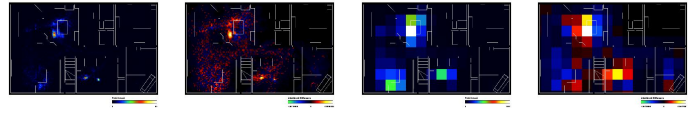
"sara" Utterances: 47 AoA: 23.4



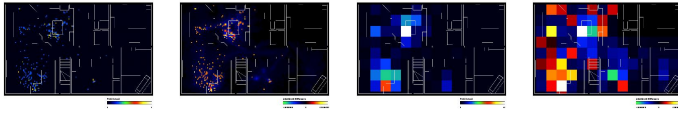
"saw" Utterances: 1,003 AoA: 23.8



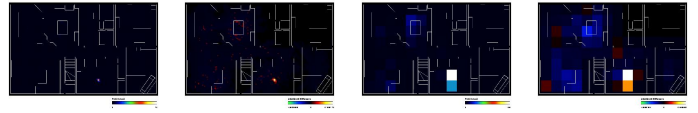
"say" Utterances: 7,757 AoA: 20.8



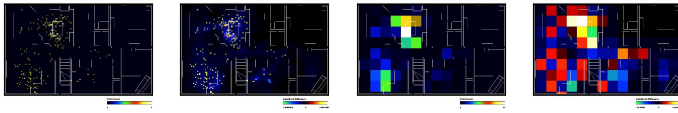
"school" Utterances: 225 AoA: 19.7



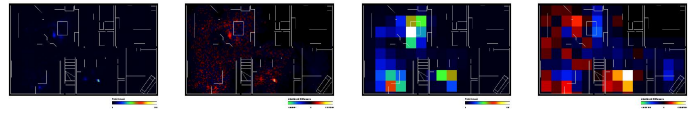
"sea" Utterances: 387 AoA: 18.8



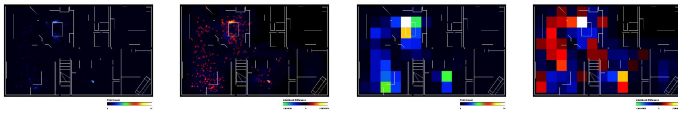
"seat" Utterances: 165 AoA: 22.1



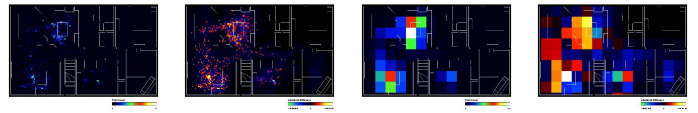
"see" Utterances: 7,764 AoA: 19.4



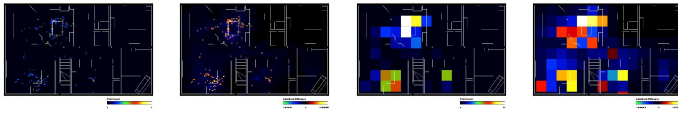
"set" Utterances: 389 AoA: 21.9



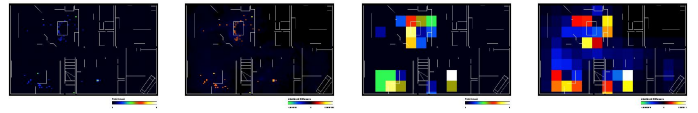
"seven" Utterances: 709 AoA: 17.7



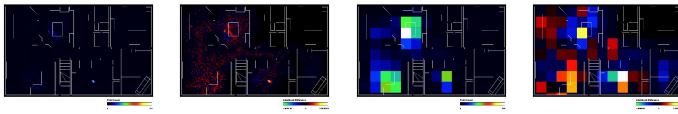
"shake" Utterances: 140 AoA: 21.1



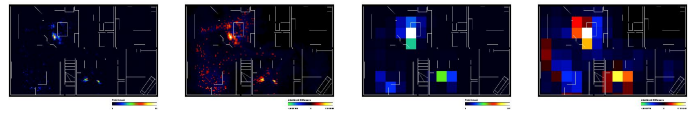
"shark" Utterances: 48 AoA: 18.0



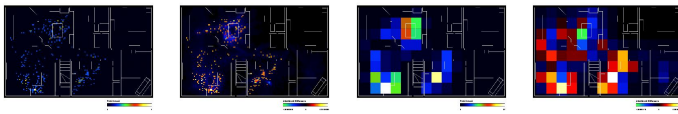
"she" Utterances: 2,847 AoA: 20.8



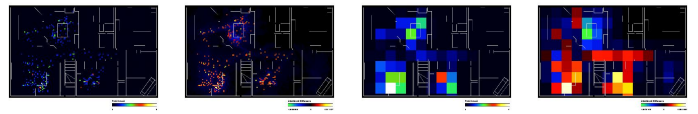
"sheep" Utterances: 954 AoA: 15.4



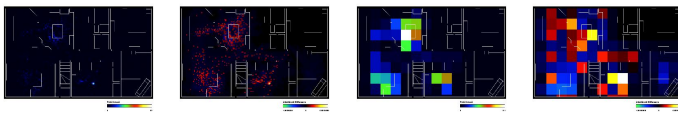
"shirt" Utterances: 240 AoA: 16.9



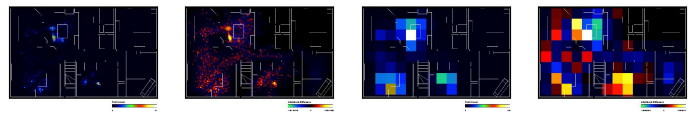
"shoe" Utterances: 274 AoA: 16.6



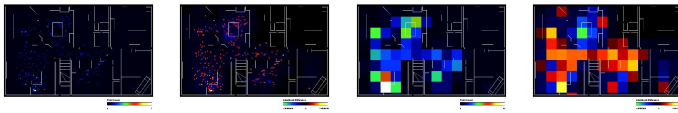
"should" Utterances: 716 AoA: 15.0



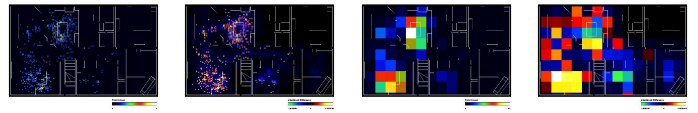
"show" Utterances: 2,128 AoA: 19.5



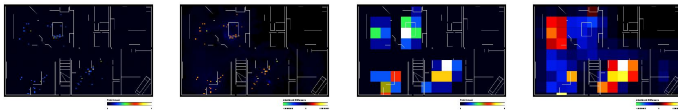
"shower" Utterances: 210 AoA: 18.9



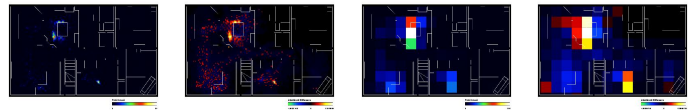
"side" Utterances: 423 AoA: 20.4



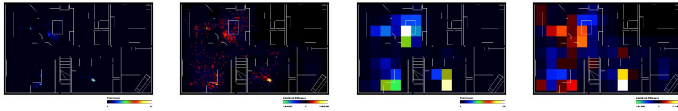
"silver" Utterances: 70 AoA: 20.2



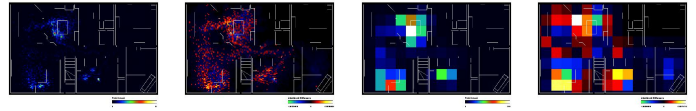
"sing" Utterances: 1,266 AoA: 19.7



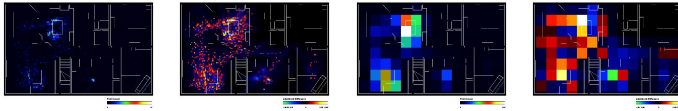
"sir" Utterances: 914 AoA: 20.7



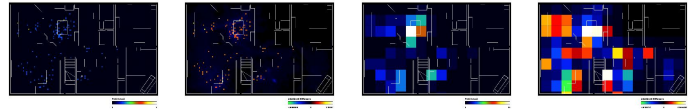
"sit" Utterances: 1,817 AoA: 18.8



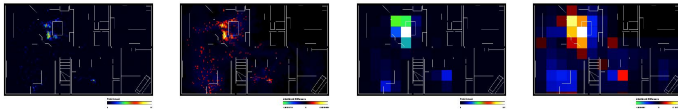
"six" Utterances: 1,237 AoA: 22.8



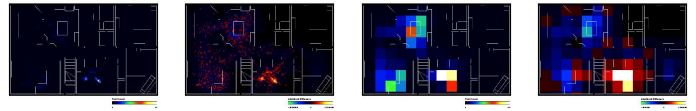
"skin" Utterances: 98 AoA: 21.1



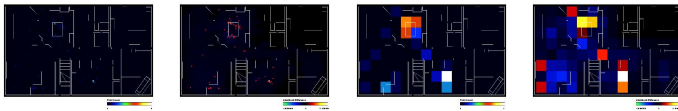
"sky" Utterances: 691 AoA: 17.7



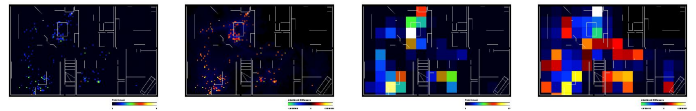
"sleep" Utterances: 1,243 AoA: 17.6



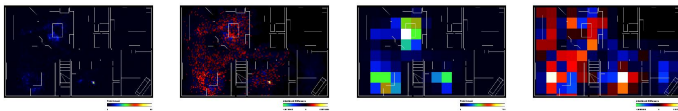
"small" Utterances: 53 AoA: 11.0



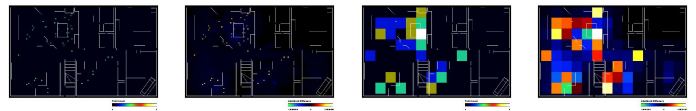
"snow" Utterances: 178 AoA: 18.9



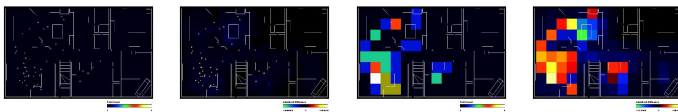
"so" Utterances: 4,772 AoA: 16.1



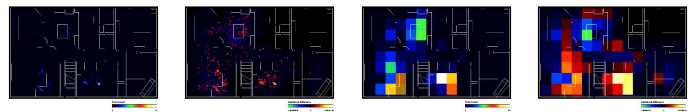
"soap" Utterances: 41 AoA: 21.7



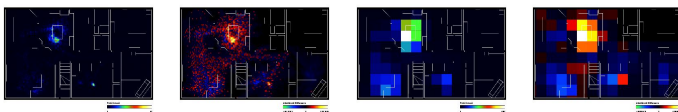
"soccer" Utterances: 38 AoA: 20.0



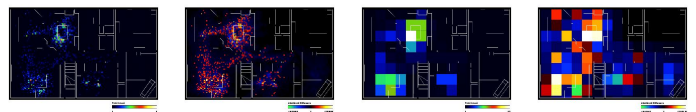
"socks" Utterances: 334 AoA: 17.5



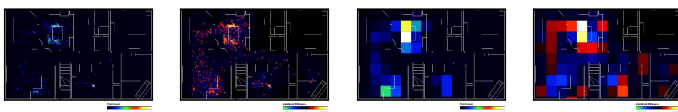
"some" Utterances: 4,514 AoA: 18.4



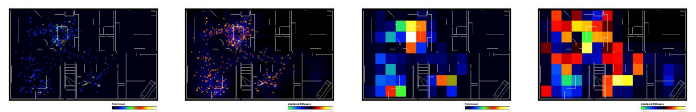
"something" Utterances: 944 AoA: 15.8



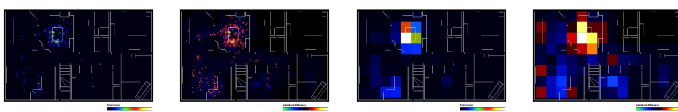
"song" Utterances: 572 AoA: 22.2



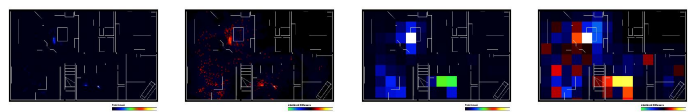
"sorry" Utterances: 348 AoA: 16.4



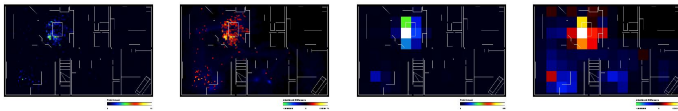
"soup" Utterances: 260 AoA: 21.3



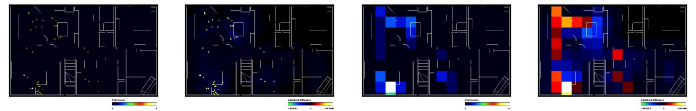
"spider" Utterances: 1,067 AoA: 22.6



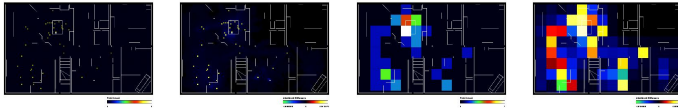
"spoon" Utterances: 422 AoA: 16.7



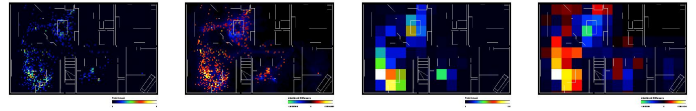
"squirrel" Utterances: 41 AoA: 19.3



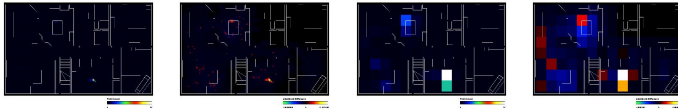
"stairs" Utterances: 31 AoA: 19.7



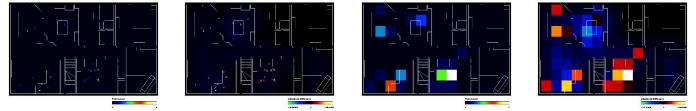
"stand" Utterances: 726 AoA: 20.4



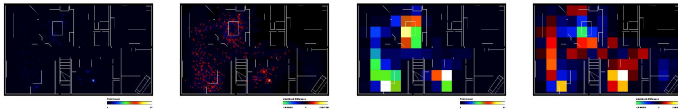
"star" Utterances: 176 AoA: 13.1



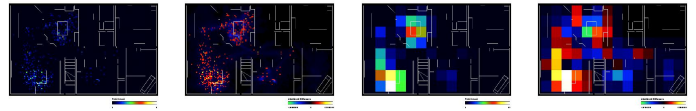
"starfish" Utterances: 46 AoA: 19.7



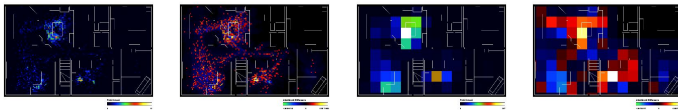
"stay" Utterances: 525 AoA: 20.4



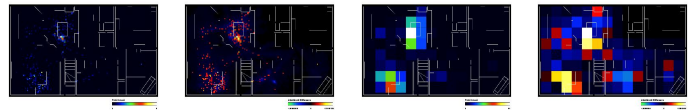
"stick" Utterances: 361 AoA: 20.9



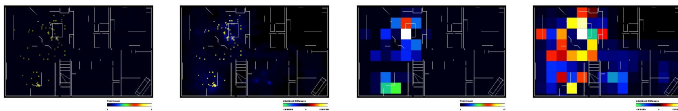
"stop" Utterances: 1,543 AoA: 21.9



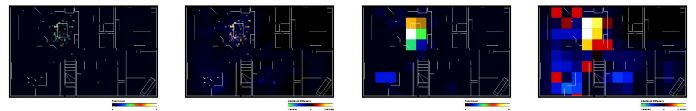
"store" Utterances: 281 AoA: 20.7



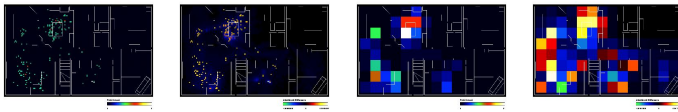
"straw" Utterances: 59 AoA: 19.4



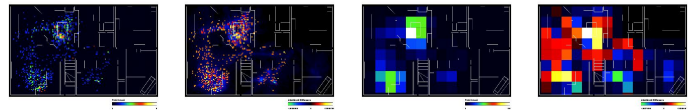
"strawberry" Utterances: 59 AoA: 18.8



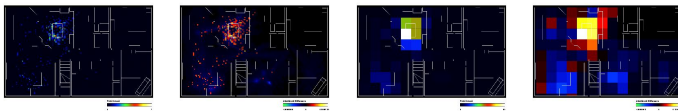
"stuck" Utterances: 131 AoA: 17.6



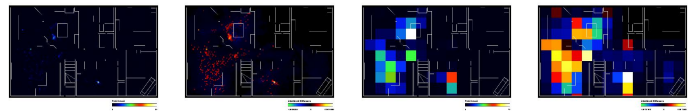
"stuff" Utterances: 655 AoA: 17.5



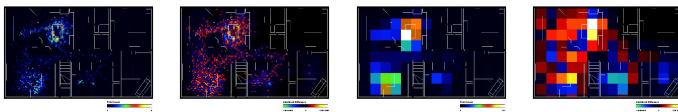
"sugar" Utterances: 239 AoA: 18.8



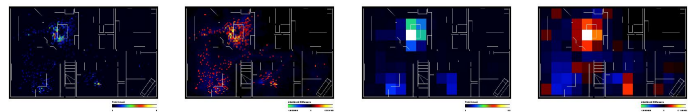
"sun" Utterances: 517 AoA: 15.0



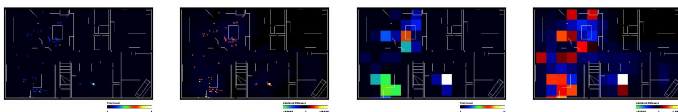
"sure" Utterances: 1,319 AoA: 21.8



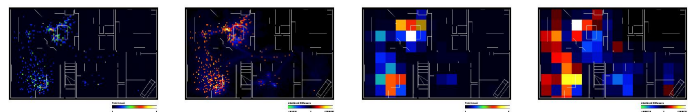
"sweet" Utterances: 634 AoA: 23.0



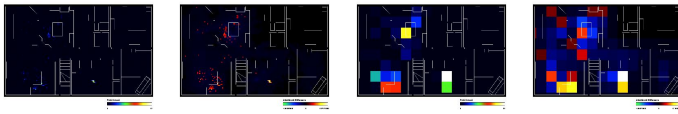
"swimming" Utterances: 77 AoA: 19.7



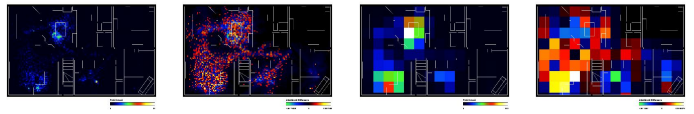
"table"	Utterances: 422	AoA: 18.5
---------	-----------------	-----------



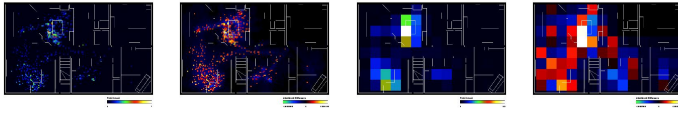
"tail" Utterances: 198 AoA: 20.4



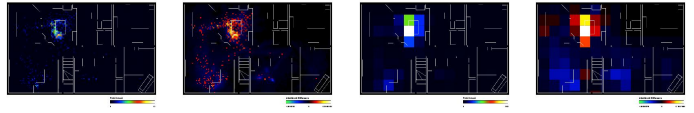
"take" Utterances: 3,105 AoA: 20.1



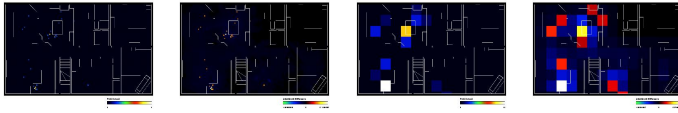
"talk" Utterances: 655 AoA: 20.4



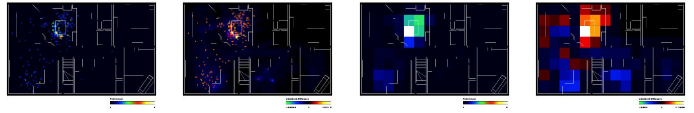
"taste" Utterances: 510 AoA: 20.7



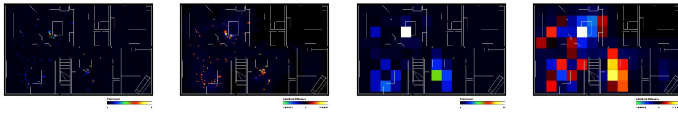
Word	Utterances	AoA
"taxi"	30	18.3



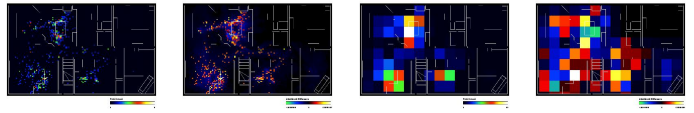
"tea" Utterances: 170 AoA: 17.7



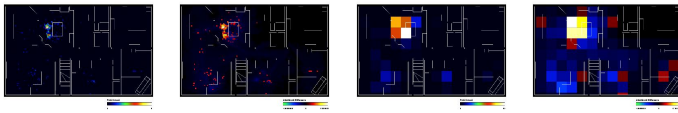
"teddy" Utterances: 122 AoA: 20.4



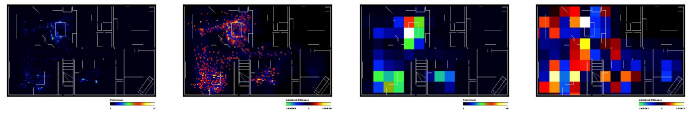
"teeth" Utterances: 530 AoA: 20.8



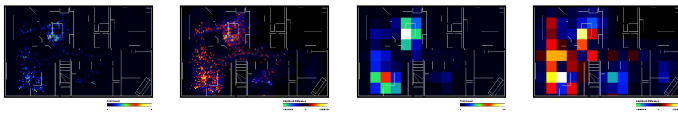
"telephone" Utterances: 240 AoA: 19.9



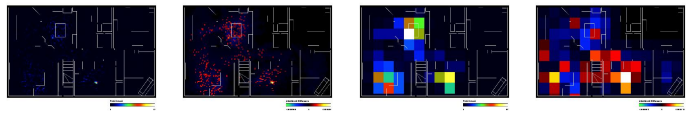
"tell" Utterances: 790 AoA: 17.9



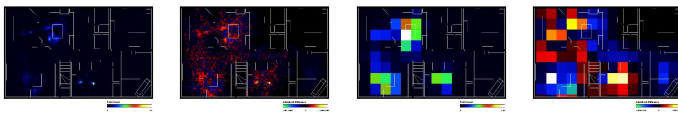
"ten" Utterances: 742 AoA: 19.1



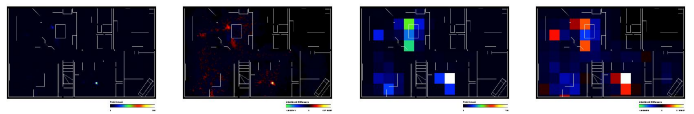
"thank" Utterances: 449 AoA: 15.6



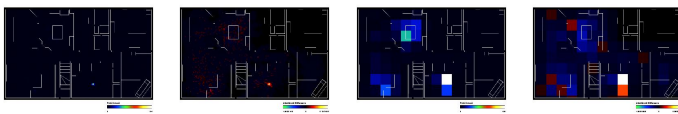
"that" Utterances: 7,863 AoA: 14.5



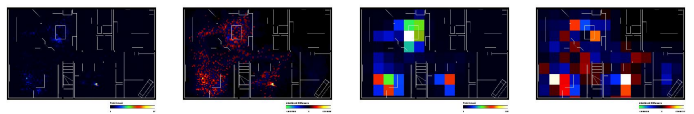
"the" Utterances: 2,883 AoA: 10.3



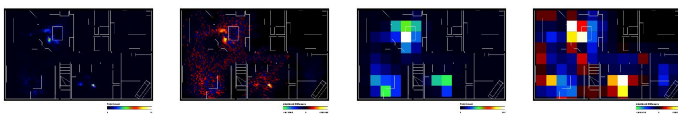
"them" Utterances: 1,257 AoA: 15.0



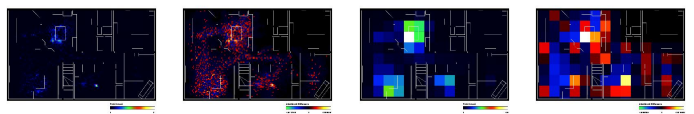
"then" Utterances: 1,665 AoA: 15.0



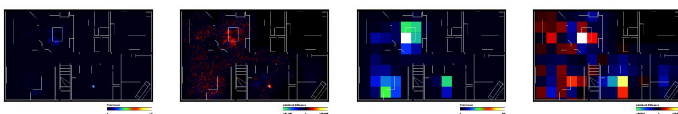
"there" Utterances: 4,868 AoA: 16.1



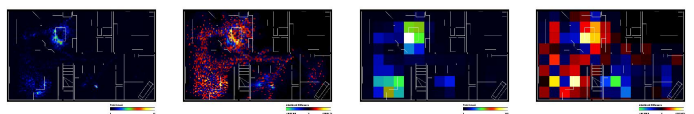
"these" Utterances: 2,174 AoA: 23.3



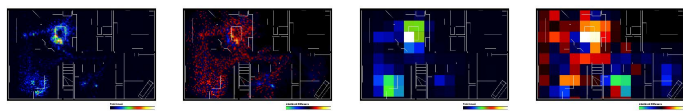
"they" Utterances: 4,196 AoA: 20.1



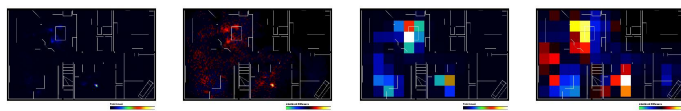
"thing" Utterances: 4,526 AoA: 22.7



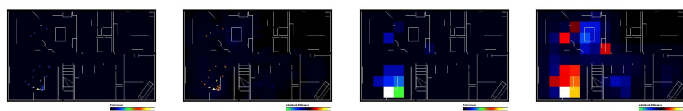
"think" Utterances: 4,798 AoA: 20.4



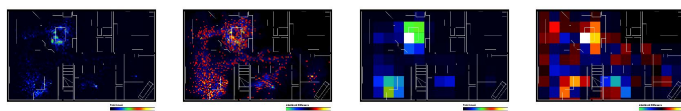
"this" Utterances: 6,034 AoA: 14.4



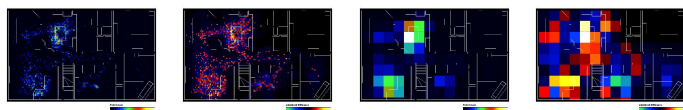
"thomas" Utterances: 54 AoA: 21.4



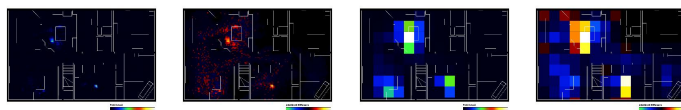
"those" Utterances: 1,967 AoA: 22.9



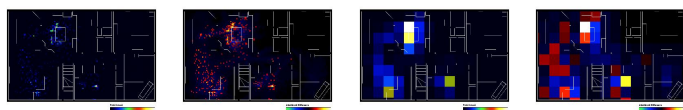
"though" Utterances: 1,037 AoA: 22.8



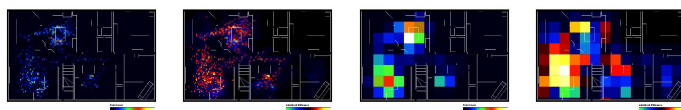
"three" Utterances: 2,083 AoA: 16.2



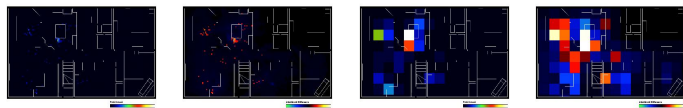
"through" Utterances: 516 AoA: 16.4



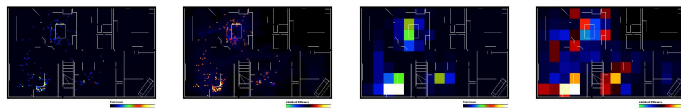
"throw" Utterances: 828 AoA: 16.9



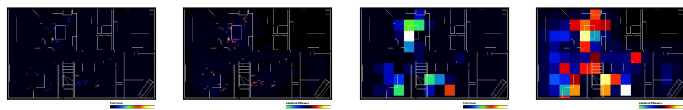
"thumper" Utterances: 67 AoA: 20.2



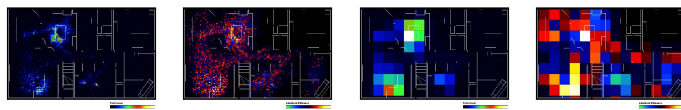
"tickle" Utterances: 195 AoA: 18.7



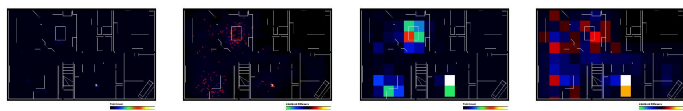
"tiger" Utterances: 63 AoA: 18.7



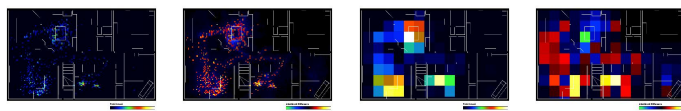
"time" Utterances: 4,424 AoA: 23.4



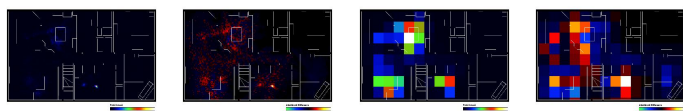
"tiny" Utterances: 230 AoA: 20.4



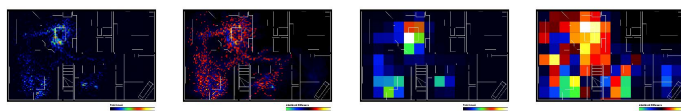
"tired" Utterances: 570 AoA: 20.4



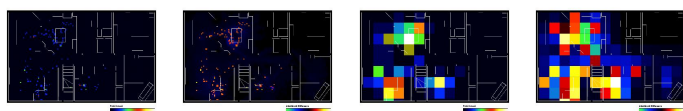
"to" Utterances: 7,304 AoA: 13.9



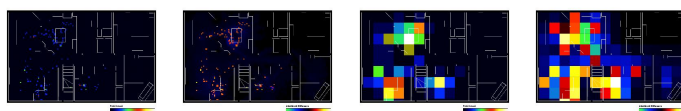
"today" Utterances: 1,588 AoA: 17.6



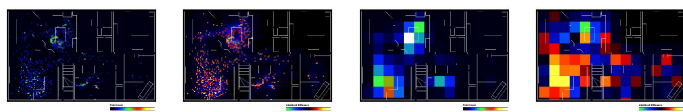
"toe" Utterances: 130 AoA: 16.9



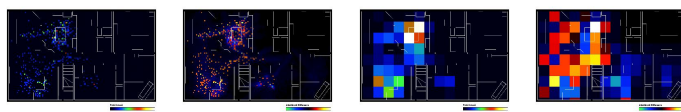
"toes" Utterances: 130 AoA: 16.9



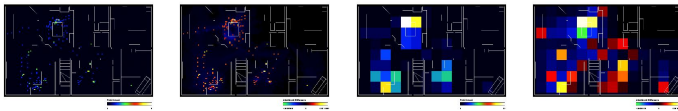
"together" Utterances: 680 AoA: 24.1



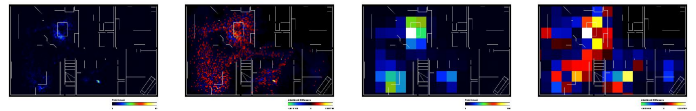
"tomorrow" Utterances: 305 AoA: 16.5



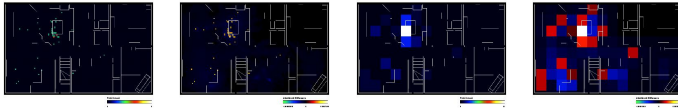
"tongue" Utterances: 187 AoA: 17.5



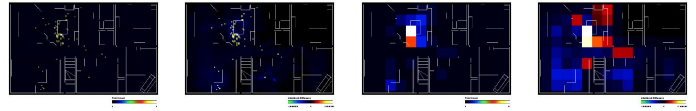
"too" Utterances: 2,913 AoA: 20.4



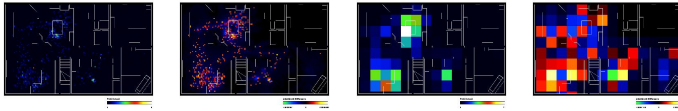
"toothbrush" Utterances: 35 AoA: 18.5



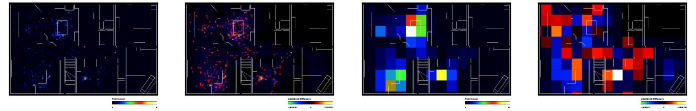
"toothpaste" Utterances: 46 AoA: 20.7



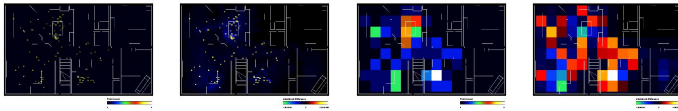
"top" Utterances: 455 AoA: 22.7



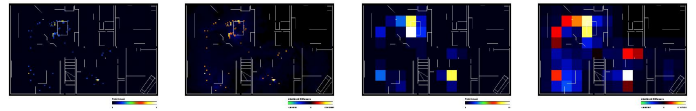
"touch" Utterances: 369 AoA: 18.0



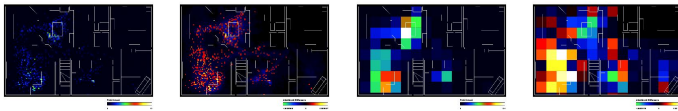
"towel" Utterances: 78 AoA: 18.9



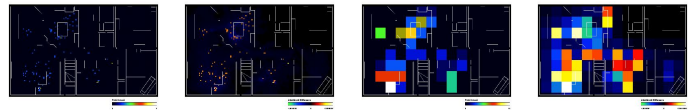
"town" Utterances: 88 AoA: 15.0



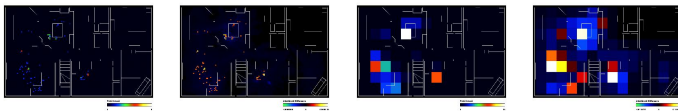
"toy" Utterances: 713 AoA: 19.3



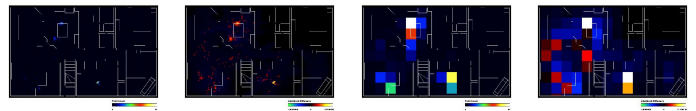
"track" Utterances: 86 AoA: 17.5



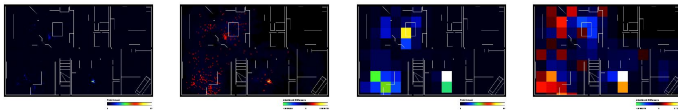
"tractor" Utterances: 84 AoA: 19.4



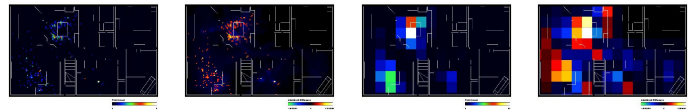
"train" Utterances: 408 AoA: 15.5



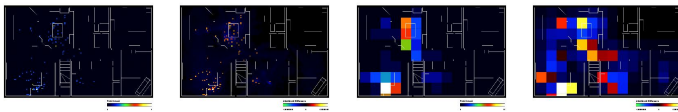
"tree" Utterances: 467 AoA: 16.8



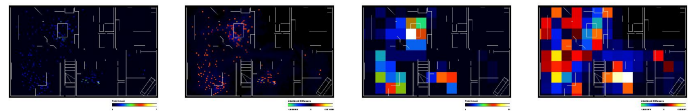
"triangle" Utterances: 298 AoA: 19.2



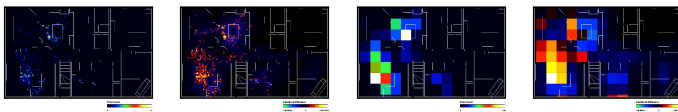
"[name 3]" Utterances: 85 AoA: 19.5



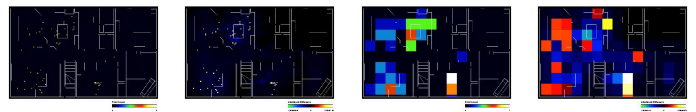
"trouble" Utterances: 176 AoA: 20.0



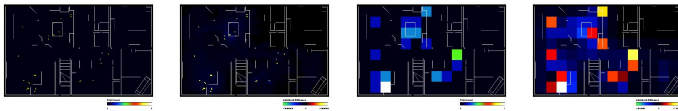
"truck" Utterances: 732 AoA: 14.8



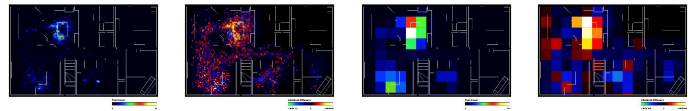
"true" Utterances: 49 AoA: 14.2



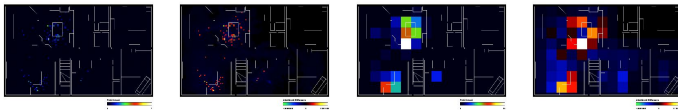
"trunk" Utterances: 24 AoA: 17.7



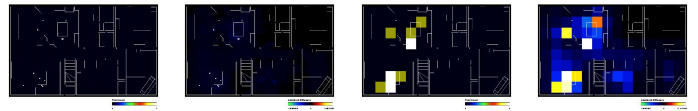
"try" Utterances: 3,018 AoA: 20.8



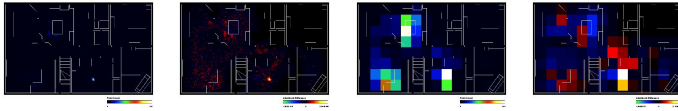
"tummy" Utterances: 90 AoA: 19.9



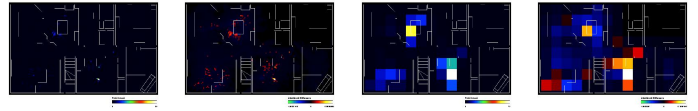
"tunnel" Utterances: 10 AoA: 18.9



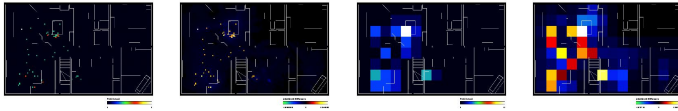
"turn" Utterances: 2,269 AoA: 23.3



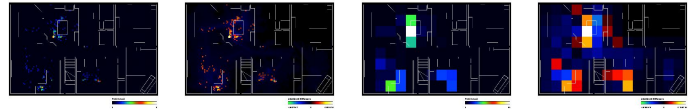
"turtle" Utterances: 302 AoA: 18.4



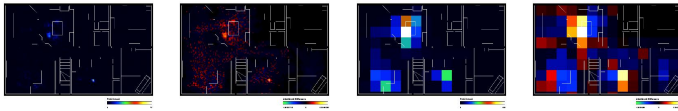
"tweet" Utterances: 48 AoA: 20.3



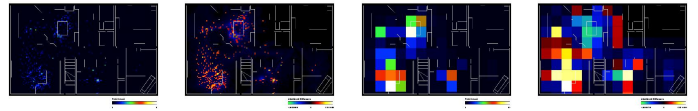
"twinkle" Utterances: 224 AoA: 19.9



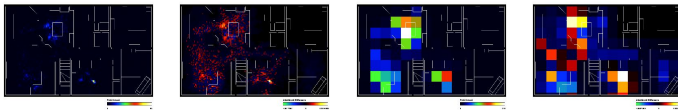
"two" Utterances: 2,460 AoA: 16.5



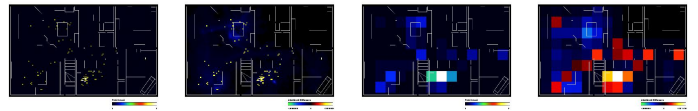
"under" Utterances: 379 AoA: 20.4



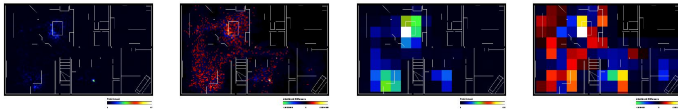
"up" Utterances: 2,619 AoA: 13.8



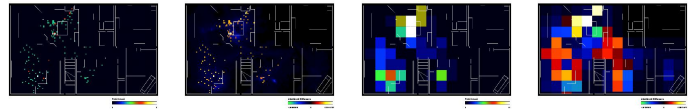
"vaseline" Utterances: 67 AoA: 20.2



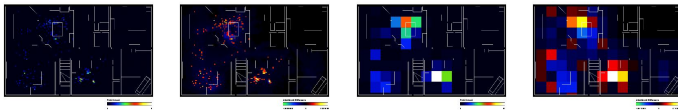
"very" Utterances: 2,005 AoA: 21.0



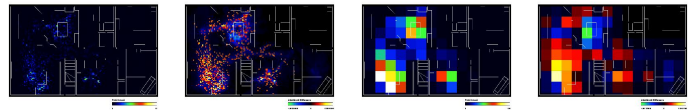
"vroom" Utterances: 135 AoA: 15.5



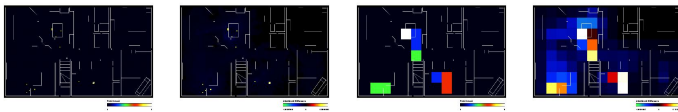
"wait" Utterances: 250 AoA: 11.7



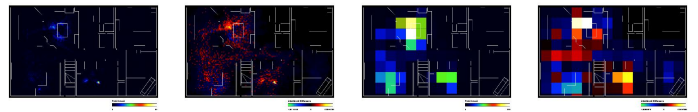
"walk" Utterances: 968 AoA: 18.7



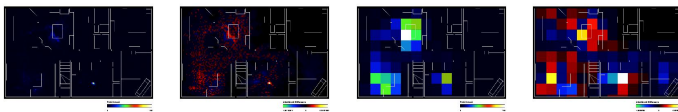
"walrus" Utterances: 19 AoA: 21.5



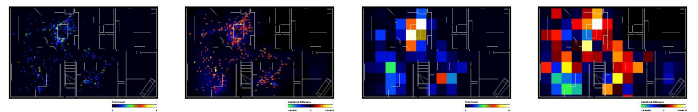
"want" Utterances: 3,523 AoA: 13.1



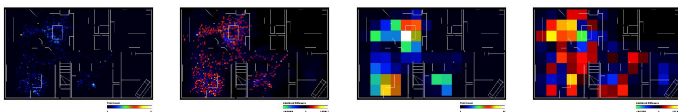
"was" Utterances: 4,251 AoA: 16.5



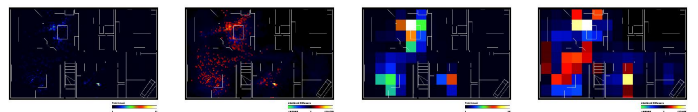
"wash" Utterances: 341 AoA: 19.4



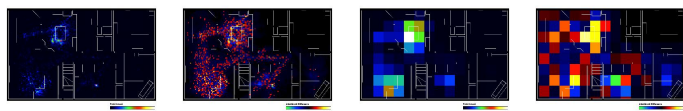
"watch" Utterances: 747 AoA: 19.4



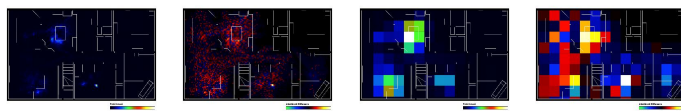
"water" Utterances: 791 AoA: 13.0



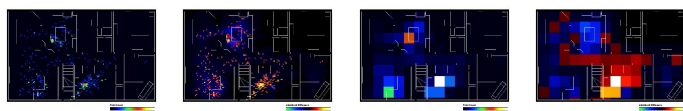
"way" Utterances: 2,096 AoA: 22.1



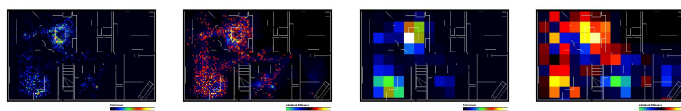
"we" Utterances: 12,197 AoA: 20.2



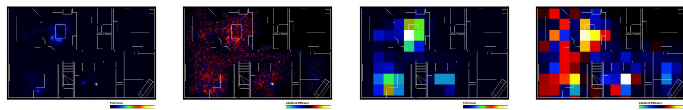
"wear" Utterances: 422 AoA: 21.9



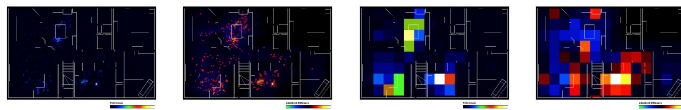
"well" Utterances: 1,424 AoA: 17.7



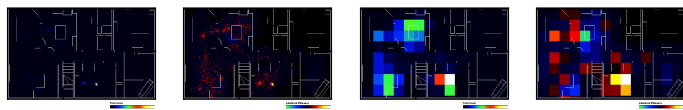
"were" Utterances: 12,197 AoA: 20.2



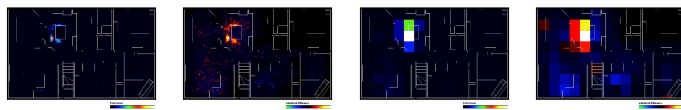
"wet" Utterances: 381 AoA: 21.2



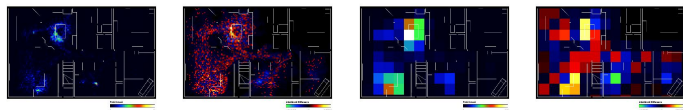
"what" Utterances: 1,180 AoA: 10.6



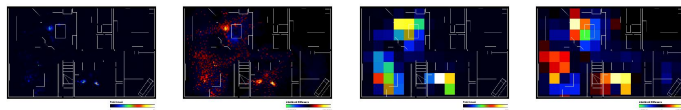
"wheel" Utterances: 900 AoA: 17.7



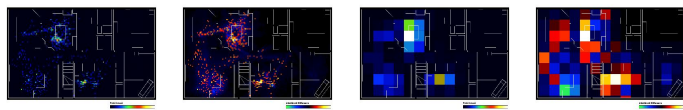
"when" Utterances: 4,919 AoA: 24.1



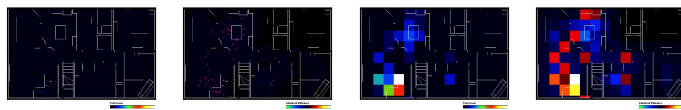
"where" Utterances: 2,869 AoA: 13.6



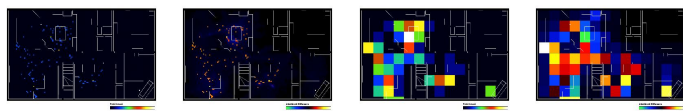
"which" Utterances: 612 AoA: 15.3



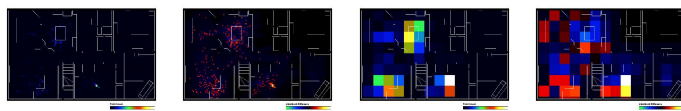
"whine" Utterances: 53 AoA: 17.5



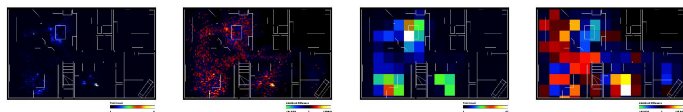
"whistle" Utterances: 109 AoA: 21.3



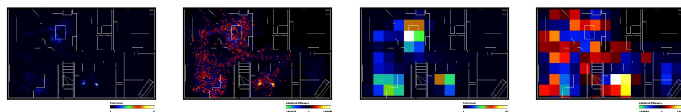
"white" Utterances: 375 AoA: 17.7



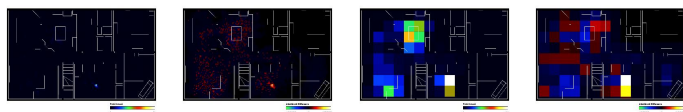
"who" Utterances: 2,996 AoA: 20.4



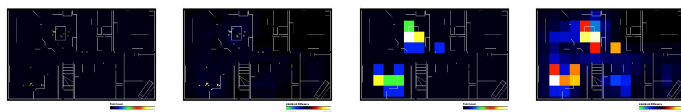
"why" Utterances: 1,821 AoA: 16.5



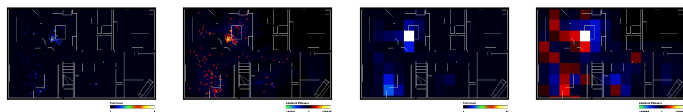
"will" Utterances: 1,044 AoA: 16.2



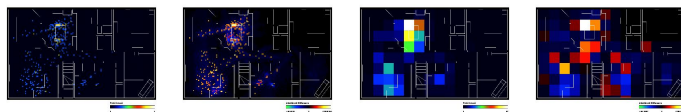
"windmill" Utterances: 11 AoA: 22.2



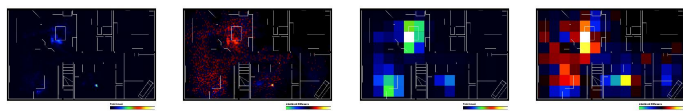
"window" Utterances: 223 AoA: 19.4



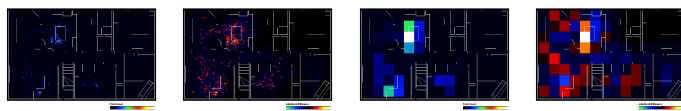
"wipe" Utterances: 288 AoA: 21.7



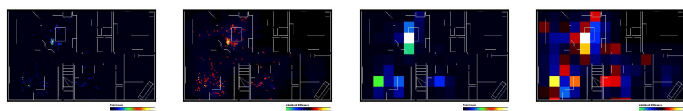
"with" Utterances: 7,689 AoA: 20.3



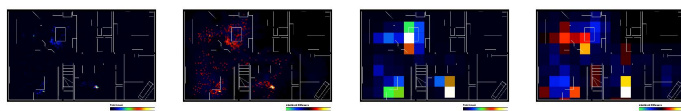
"wonder" Utterances: 319 AoA: 20.1



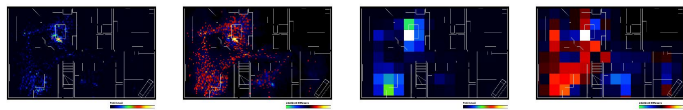
"woof" Utterances: 296 AoA: 18.0



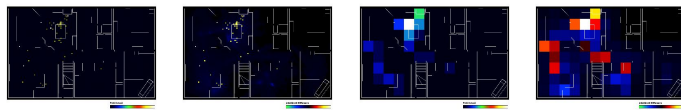
"wool" Utterances: 712 AoA: 21.0



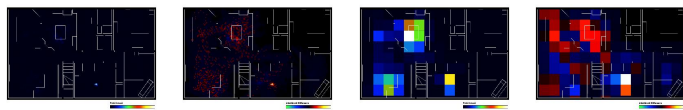
"work" Utterances: 1,325 AoA: 20.8



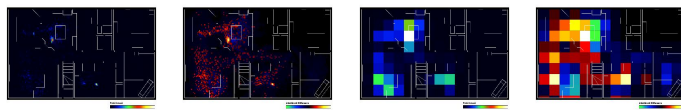
"wormy" Utterances: 48 AoA: 21.7



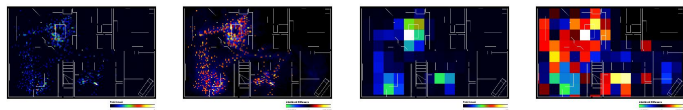
"would" Utterances: 1,196 AoA: 17.6



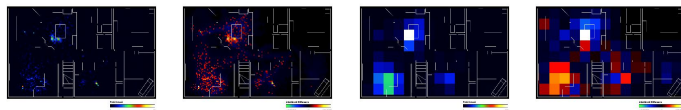
"wow" Utterances: 1,440 AoA: 15.2



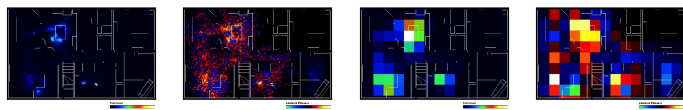
"wrong" Utterances: 692 AoA: 18.7



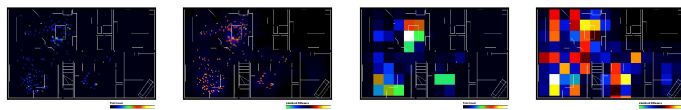
"yellow" Utterances: 869 AoA: 18.1



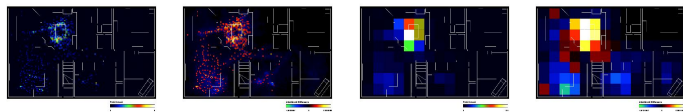
"yes" Utterances: 14,555 AoA: 16.1



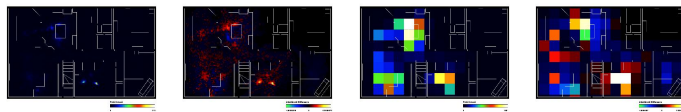
"yet" Utterances: 193 AoA: 15.2



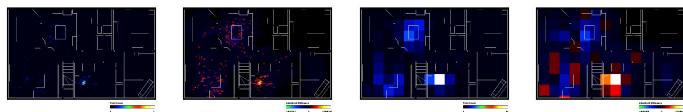
"yogurt" Utterances: 610 AoA: 17.6



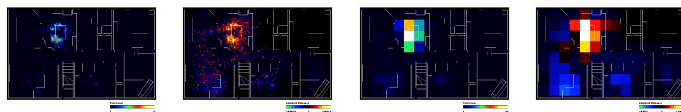
"you" Utterances: 14,524 AoA: 12.7



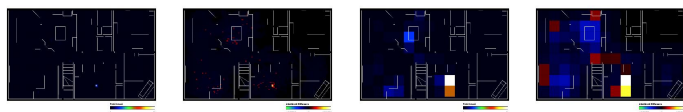
"yuck" Utterances: 289 AoA: 16.7



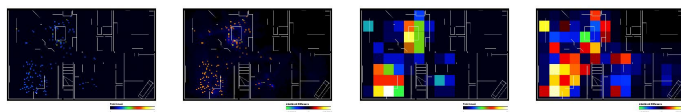
"yum" Utterances: 1,147 AoA: 18.7



"zoo" Utterances: 119 AoA: 16.8



"zoom" Utterances: 104 AoA: 21.5



Bibliography

- [1] K Bernardin and R Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 5, 2008.
- [2] Jerome Bruner. The role of interaction formats in language acquisition. In J.P. Forgas, editor, *Language and social situations*, pages 31–46. Springer, 1985.
- [3] D Comaniciu and P Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on pattern analysis and machine intelligence*, pages 603–619, 2002.
- [4] P DeCamp. Headlock: wide-range head pose estimation for low resolution video (2007). wide-range head pose estimation for low resolution video. *en.scientificcommons.org*.
- [5] P DeCamp and D Roy. A human-machine collaborative approach to tracking human movement in multi-camera video. *Proceeding of the ACM International Conference on Image and Video Retrieval*, pages 1–8, 2009.
- [6] P DeCamp, G Shaw, R Kubat, and D Roy. An immersive system for browsing and visualizing surveillance video. *Proceedings of the international conference on Multimedia*, pages 371–380, 2010.
- [7] P.M Dixon. Ripley’s k function. 2002.
- [8] R Duval. Representation, vision and visualization: Cognitive functions in mathematical thinking. basic issues for learning. 1999.
- [9] Michael Friendly. Milestones in the history of thematic cartography, statistical graphics, and data visualization. pages 1–79, Aug 2009.
- [10] N Funk. A study of the kalman filter applied to visual tracking. *University of Alberta*, 2003.
- [11] P Haase. Spatial pattern analysis in ecology based on ripley’s k-function: introduction and methods of edge correction. *Journal of Vegetation Science*, 6(4):575–582, 1995.

- [12] Y Ivanov, A Sorokin, C Wren, and I Kaur. Tracking people in mixed modality systems. *Visual Communications and Image Processing, volume EI123. IS&T/SPIE*, 2007.
- [13] JCR Licklider. Man-computer symbiosis. *Human Factors in Electronics, IRE Transactions on*, (1):4–11, 2008.
- [14] T Lochmatter, P Roduit, C Cianci, N Correll, J Jacot, and A Martinoli. Swistrack-a flexible open source tracking software for multi-agent systems. *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 4004–4010, 2008.
- [15] DG Lowe. Object recognition from local scale-invariant features. *iccv*, page 1150, 1999.
- [16] J.M McHugh, J Konrad, V Saligrama, and P.M Jodoin. Foreground-adaptive background subtraction. *Signal Processing Letters, IEEE*, 16(5):390–393, 2009.
- [17] B Milch. Artificial general intelligence through large-scale, multimodal bayesian learning. *Proceeding of the 2008 conference on Artificial General Intelligence 2008: Proceedings of the First AGI Conference*, pages 248–255, 2008.
- [18] Matt Miller. Whodat. 2010.
- [19] Matt Miller. Semantic spaces: Behavior, language and word learning in the human speechome corpus. 2011.
- [20] P.A.P Moran. Notes on continuous stochastic phenomena. *Biometrika*, 37(1/2):17–23, 1950.
- [21] D Parkhurst, K Law, and E Niebur. Modeling the role of salience in the allocation of overt visual attention. *Vision research*, 42(1):107–123, 2002.
- [22] F.H Post, G.M Nielson, and G.P Bonneau. *Data visualization: the state of the art*, volume 1. 2003.
- [23] ZW Pylyshyn. Situating vision in the world. *Trends in Cognitive Sciences*, 4(5):197–207, 2000.
- [24] S Rao. Visual routines and attention. 2002.
- [25] Brandon C. Roy. Bounds on the expected entropy and kl-divergence of sampled multinomial distributions. 2011.
- [26] D Roy. Grounding words in perception and action: computational insights. *Trends in cognitive sciences*, 9(8):389–396, 2005.
- [27] D Roy and B.C Roy. Human-machine collaboration for rapid speech transcription. 2007.

- [28] Deb Roy, Rupal Patel, Philip DeCamp, Rony Kubat, Michael Fleischman, Brandon Roy, Nikolaos Mavridis, Stefanie Tellex, Alexia Salata, Jethran Guinness, Michael Levit, and Peter Gorniak. The human speechome project. *The 28th Annual Conference of the Cognitive Science Society*, pages 1–6, Feb 2006.
- [29] J Rymel, J Renno, D Greenhill, J Orwell, and GA Jones. Adaptive eigen-backgrounds for object detection. *Image Processing, 2004. ICIP'04. 2004 International Conference on*, 3:1847–1850 Vol. 3, 2004.
- [30] BJ Scholl, ZW Pylyshyn, and J Feldman. What is a visual object? evidence from target merging in multiple object tracking. *Cognition*, 80(1-2):159–177, 2001.
- [31] George M. Shaw. Multi-modal classification of tracking output for person identification. pages 1–5, Dec 2009.
- [32] ES Spelke. Principles of object perception. *Cognitive Science: A Multidisciplinary Journal*, 14(1):29–56, 1990.
- [33] C Stauffer and WEL Grimson. Adaptive background mixture models for real-time tracking. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:246–252, 1999.
- [34] G Tononi. Consciousness as integrated information: a provisional manifesto. *The Biological Bulletin*, 215(3):216, 2008.
- [35] Edward R. Tufte. *The Visual Display of Quantitative Information*. 1983.
- [36] A Veeraraghavan and R Chellappa. Tracking bees in a hive. 2008.
- [37] S Vosoughi, B.C Roy, M.C Frank, and D Roy. Contributions of prosodic and distributional features of caregivers’ speech in early word learning. *Proceedings of the 32nd Annual Cognitive Science Conference*, 2010.
- [38] Wikipedia. The hungarian algorithm, 2011.
- [39] Wikipedia. Ramer-douglas-peucker algorithm, 2011.
- [40] M.P Winsor. Taxonomy was the foundation of darwin’s evolution. *Taxon*, 58(1):43–49, 2009.
- [41] A Yilmaz, O Javed, and M Shah. Object tracking: A survey. *ACM Computing Surveys (CSUR)*, 38(4):13, 2006.