The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Placing Objects in Gesture Space: Toward Incremental Interpretation of Multimodal Spatial Descriptions

Ting Han,[1] Casey Kennington,[2] David Schlangen[1]
[1]Dialogue Systems Group // CITEC, Bielefeld University, [2]Boise State University
{ting.han, david.schlangen}@uni-bielefeld.de, caseykennington@boisestate.edu

Abstract

When describing routes not in the current environment, a common strategy is to anchor the description in configurations of salient landmarks, complementing the verbal descriptions by "placing" the non-visible landmarks in the gesture space. Understanding such multimodal descriptions and later locating the landmarks from real world is a challenging task for the hearer, who must interpret speech and gestures in parallel, fuse information from both modalities, build a mental representation of the description, and ground the knowledge to real world landmarks. In this paper, we model the hearer's task, using a multimodal spatial description corpus we collected. To reduce the variability of verbal descriptions, we simplified the setup to use simple objects as landmarks. We describe a real-time system to evaluate the separate and joint contributions of the modalities. We show that gestures not only help to improve the overall system performance, even if to a large extent they encode redundant information, but also result in earlier final correct interpretations. Being able to build and apply representations incrementally will be of use in more dialogical settings, we argue, where it can enable immediate clarification in cases of mismatch.

Introduction

Navigating robots with natural route descriptions is a goal researchers have long aspired to. While good performances have been achieved on interpreting pure language navigations (e.g., "*near the second door on your right*") and multimodal commands in the shared space (e.g., *"go over there+[hand pointing to the destination]"*) (Kollar et al. 2010; Skubic et al. 2004; Williams et al. 2016), not much attention has been paid to the scenario of understanding multimodal route descriptions when the described route is not in current view.

In such cases, anchoring the destination in configurations of landmarks and indicating their relative spatial layout with deictic gestures pointing into the empty gestural space is a common practice (Emmorey, Tversky, and Taylor 2000; Alibali 2005; Cassell et al. 2007). For example, to help the listener identify a hotel in a busy town centre, a speaker might produce a multimodal description like the following:

(1) Here$_{[deixis]}$ is the bus stop, a bit left of it$_{[deixis]}$ is a church and right in front of that$_{[deixis]}$ is the hotel.

Figure 1: Providing a multimodal description (*left*) of a scene (*right*).

The deictic gestures map the spatial layout of the landmarks from the speaker's mental image to the shared gesture space (McNeill 1992). Together with the verbal descriptions, a listener can build a mental representation of the landmarks, later navigating itself by comparing the mental representation with real-world landmarks.

While the verbal descriptions provide important information about the denoted objects (e.g., entity name: *the bus stop*, relative position: *a bit left of*), the deictic gestures complement the verbal content with spatial information (i.e., points with coordinates in the gesture space, standing in for the real locations of the referents, and indicating their spatial relation). Only together do gestures and words receive a definite meaning (e.g., how much *left* is *a bit left*, relative to *below*). Hence, the resolution task in this setting goes beyond previous works in which gestures can be grounded to objects present in the environment (Stiefelhagen et al. 2004).

Psycholinguistic studies show that humans process gestures and speech jointly and incrementally (Campana et al. 2005). While descriptions unfold, listeners immediately integrate information from co-occurring speech and gestures. Moreover, to apply they interpretation later, it's essential to form a hypothesis in mind, making it a very demanding cognitive, language-related tasks (Schneider and Taylor 1999).

In this paper, we model the joint and incremental interpretation of multimodal spatial descriptions, using a simplified spatial description task. Specifically, we address following questions: 1) to what degree can deictic gestures improve the interpretation accuracy; 2) how gestures benefit the interpretation of spatial descriptions on the incremental level. We collected a multimodal spatial description corpus which was elicited with a simplified scene description task (see details in **Data collection**). The corpus includes hand motion and natural verbal descriptions of objects in simple scenes. With the collected data, we built a real-time system that concurrently processes speech and gestures, where deictic gestures are treated as denoting attributes of and relations between referents. The results show that as compared to using only language information, incorporating gestures enables more accurate understanding of the descriptions and earlier final correct retrieving decision. The corpus is publicly available.[1]

Related work

Enabling robots to understand navigation instructions and spatial relations is an active research topic. Previous works have shown that natural language is efficient for constructing spatial-semantic representations of known environments (Kollar et al. 2010; Hemachandra et al. 2014; Boularias et al. 2016; Mei, Bansal, and Walter 2016; Tellex et al. 2011). These works require a semantically labeled representation of the environment. A more challenging task is to understand instructions in an unknown environment. To understand such instructions, a robot/agent needs to ground natural language instructions to the situated environment and conjecture spatial relations between entities which are potentially unknown to the robot (Duvallet, Kollar, and Stentz 2013; Williams et al. 2016; 2016; 2015; Duvallet et al. 2016). In this work, we aim to build a system that can understand multimodal spatial descriptions of unknown environment. The work goes beyond previous works by incorporating abstract deictic gestures.

Interpreting multimodal navigation instructions has been widely studied. Most previous works focus on understanding navigations in situated environment. (Skubic et al. 2004) described a multimodal interface which understands spatial relations using natural language, deictic and demonstrative gestures as input. (Stiefelhagen et al. 2004) focuses on interpreting spontaneous reference expression (speech, pointing and head gestures) when interacting with the situated environment, rather than interpreting spatial descriptions not in the environment. (Whitney et al. 2016) proposed a Bayesian model to continuously understand pointing gestures and language referential expressions. The approach was evaluated with a final referential accuracy. (Matuszek et al. 2014) presented a high-level architecture of interpreting unscripted deictic gesture and natural language for human-robot interactions, in which all referents are represented to the robot. In this paper, we investigate how deictic gestures benefit interpretations on the incremental level. We found that besides a higher final accuracy, incrementally interpreting deictic gestures also leads to an earlier final correct decision.

Data collection

Task description

We designed a simple scene description task to elicit natural multimodal descriptions, as shown in Figure 1. To focus on the natural and incremental nature of the descriptions, we designed in a somewhat idealised setting (similar to (Roy 2002)), replacing landmarks as they would appear on a real-world map with simple geometric shapes. This is intended to reduce the cognitive load of participants and variance of the verbal descriptions, while keeping the spatial complexity.

We generated 100 such scenes, each composed of two circles and a square. The *size*, *shape* and *colour* of each object were randomly selected when the scenes were generated. Object sizes are evenly distributed between 0.05 and 0.5 (as ratio to the image scene size). There are 6 colours and 2 shapes. Each of them had equal chance to be assigned to an object. The object positions were randomly generated and adjusted until none of the objects overlap with other objects.

For each description episode, a scene was displayed on a computer screen. Participants were asked to describe it verbally (spoken; in German), possibly also using deictic gestures. *Colour*, *shape*, *size* and *relative spatial relations* between the objects were suggested to be described. To provide some feedback to the participants and create some impression of interactivity (rather than of making a passive recording), after each description a score was shown on the screen, ostensibly reflecting the degree of understanding of an automated system. In reality, the score was given by a confederate who had the instruction to reward when all attribute types were mentioned.

Recording setup

In the experiment, hand motion was tracked by a Leap sensor, a USB peripheral device composed of two monochromatic cameras and three LED infrared sensors.[2] The hand motion data was recorded with MINT Tools (Kousidis, Pfeiffer, and Schlangen 2013). Audio and video were recorded by a camera. Timestamps were recorded in videos and hand motion data.

After introducing the task, participants were seated in front of a desk. A monitor was on the right of the desk to show scenes. A Leap sensor was on the desk in front of participants to track hand motion. Due to the small effective tracking area of the sensor (about 600 mm above and 250 mm to each side of the device), we set a monitor in front of participants to display hand movements. Participants were encouraged to keep their hands in the tracking area while gesturing. This helps to track all hand movements. None of them reported unnatural gestures due to the limited tracking area. Participants had several minutes to play with the sensor before the experiment so that they knew the boundaries of the effective tracking area. When the experiment started, the monitor on the right displayed a scene. Participants started to describe after watching it for a few seconds. There was no time limit for each scene description. After each description, a score was shown on the screen for 10 seconds, then the wizard advanced to the next scene. In total, 13 participants (native German speakers) took part in the experiment. Each described for 20 minutes.

Data Processing

A sample multimodal description is shown below:

[1] https://pub.uni-bielefeld.de/data/2913177

[2] www.leapmotion.com

(2) a) Hier$_{[deixis]}$ ist ein kleines Quadrat, in rot, hier$_{[deixis]}$ ist ein hellblauer kleiner Kreis und hier$_{[deixis]}$ ist ein blauer grosser Kreis.

b) *Here*$_{[deixis]}$ *is a small square, red, here*$_{[deixis]}$ *is a light blue small circle and here*$_{[deixis]}$ *is a blue big circle.*

The recordings were manually transcribed by native speakers, then temporally aligned with the recording on a word-by-word level using an automatic forced alignment approach, using the InproTK toolkit (Baumann and Schlangen 2012). Utterances for each object were manually annotated with the referred objects for training the language in NLU module (see below for more details). For example "hier ist ein kleines Quadrat" in the example above might be annotated as referring to `object1` in the scene.

The deictic gestures were manually annotated based on hand movements in videos with ELAN, a software for annotation.[3] They were also annotated with referred objects in the same way as the utterance annotation. With recorded timestamps, hand motion data were aligned with videos. Aligned data frames in hand motion data were labeled as *stroke hold* frames (hand stays at the targeted position (Kendon 1980)) or *non-stroke hold* frames (hands in movements or not refer to any target object). The annotated labels were used for training the deictic gesture detector (described in (2)).

Scene representation Each described scene was represented as a composition of three objects. Each object is represented with four attributes: colour, shape (*circle and square*), size (discretised into *small, big, medium* according to their size) and position (x, y). The position is further discretised as vertical position (*top, middle, bottom*) and horizontal position (*left, centre, right*) to be grounded to verbal descriptions (see NLU section).

Varied gesture behaviours We observed that, when describing an object, participants gestured either with one hand to denote the location or with two hands together to show the relative position between two objects. In both cases, the spatial layout of objects are encoded in gestures. The frame rate of hand motion data was around 100 frames per second as recorded by the Leap. Hence, the hand motion data is sufficient for real-time and incremental processing.

Varied verbal descriptions Although participants were strongly encouraged to mention colour, shape, size and relative positions of the objects, they were allowed to formulate descriptions in their own way. In other words, the verbal descriptions are, within these constraints, natural descriptions. The vocabulary size is 291 and participants do indeed vary in how the information was formulated and ordered: **1)** Varied colour descriptions. Participants used different words to describe the same colour. For instance, *purple* and *cyan* objects were also described as *lilac* and *light blue*, respectively; **2)** Flexible information sequence. Participants encode the information sequence as position, colour, size, shape or other sequences; **3)** Flexible gesture/speech compositions. While some participants supplement deictic gestures with position words like "*bottom left*", they also described with words like "*a bit lower*", "*here*" or simply did not encode position information in verbal descriptions.

Real-Time Multimodal Understanding System

In this section, we describe our real-time multimodal understanding system. As shown in Figure 2, the system is composed of two separate pipelines for speech and gesture processing which take speech and hand motion data as input respectively. A fusion module yields a joint interpretation by combining outputs of the two pipelines. We will first sequentially describe individual components of each pipeline, then describe how the speech and gesture information are fused and applied to a scene retrieving task.

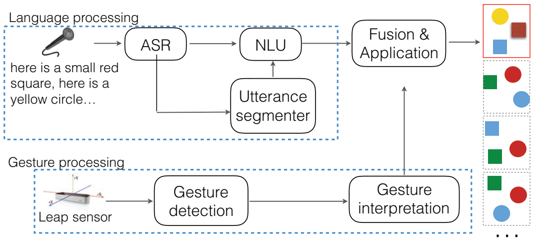

Figure 2: Multimodal system architecture.

Gesture processing: Gesture detector

The gesture detector takes hand motion data frames as input and labels each frame as *stroke hold* or *non-stroke hold*. Hence, it's a binary classifier with hand motion features as input. The training process was supervised by the annotated gesture labels. As soon as a hand motion frame is labeled as *stroke hold*, the gesture detector sends the hand position to the **Gesture interpretation** module, which interprets the gesture meaning with the *stroke-hold* coordinates.

Gesture features In the classification task, we represent each data frame with 92 raw features (as provided by the Leap SDK and recorded as above) as follows:

- **hand velocity**: the speed and movement direction of the palm in millimetres per second (3 features).
- **hand direction**: the direction from the palm position toward the fingers (3 features).
- **palm normal**: a vector perpendicular to the plane formed by the palm of the hand (3 features).
- **palm position**: the center position of the palm in millimeters from the Leap Motion Controller origin (3 features).
- **grab strength**: strength of a grab hand pose which ranges from 0 to 1(1 feature). Provided by Leap sensor.
- **finger bone directions**: the direction of finger bones (60 features).
- **finger bone angles**: "side-to-side" openness between connected finger bones (15 features). Provided by Leap sensor.
- **finger angles**: the angles between two neighbouring fingers (4 features).

While it might seem that hand velocity on its own already would give sufficient information to detect stroke hold frames, we found that sometimes participants placed hands

[3] http://tla.mpi.nl/tools/tla-tools/elan/

in the gesture space without referring to any object. Using velocity alone for classification would cause many false positive detections. We observed that, when not gesturing, hands are usually in a relaxed status with downward palms and smaller finger bone angles. Therefore, we represent each hand data frame with the above 92 features and model stroke hold recognition as a sequence classification task with a long short-term memory (LSTM) network (implemented using Keras (Chollet 2015)).

We selected a time window size of 200 ms before each frame. Every other frame was dropped in the window to reduce the load of the gesture detector. The sampled data frames compose a sequence as input for an LSTM classifier.

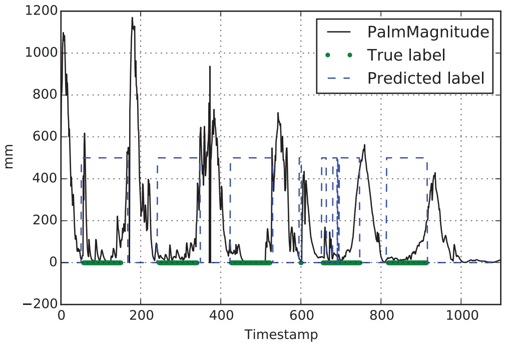

Figure 3: Stroke hold detection.

LSTM classifier The LSTM classifier includes two hidden layers and a sigmoid dense layer to give predictions. The first hidden layer has 68 nodes whose outputs are defined by *tanh* activation functions. The second hidden layer has 38 nodes and outputs via the dense layer. A dropout layer is applied to the second layer to enable more effective learning. 50% of the input units are randomly selected and set to 0 to avoid overfitting. We chose a binary cross entropy loss function optimised with a rmsprop optimiser. The training was stopped when validation loss stopped decreasing.

Figure 3 shows a sample of stroke hold detection results. As a stroke hold often lasts for several frames, we take the average hand position of all available frames, classified as being in a stroke hold, as the position for the stroke hold. As a result, we can compute and update the position of a stroke hold while it is still going on.

Gesture processing: Gesture interpretation

According to stroke hold positions detected by the Gesture detector, the gesture interpretation module resolves references of deictic gestures and evaluates how well the spatial configurations of the gestures and the objects fit with each other in a real scene, as shown in Figure 4.

Interpreting individual gestures For the 2-D scene description task in this work, we only considered the gesture position information on x and y plane and ignored the depth information along z axis. We represent the gesture space of a speaker as $\{(x, y) \in \mathbb{R}^2 : x_{min} \leq x \leq x_{max},\ y_{min} \leq y \leq y_{max}\}$, where $x_{min}, x_{max}, y_{min}, y_{max}$ are the boundaries of a speaker's gesture space estimated from all the gestures of the speaker.

As the hand movement was tracked by a Leap sensor, the stroke hold positions are defined in Leap sensor coordinate system. To compare spatial configurations between gesture and speech, gesture positions must be mapped to the image coordinate system. Given a stroke hold $(x,\ y)$, we mapped it to the image coordinate system $\{(x, y) \in \mathbb{R}^2 : 0 \leq x \leq W,\ 0 \leq y \leq H\}$, and represented the new coordinate as:

$$\mathbf{G} = \left(\frac{W * (x - x_{min})}{x_{max} - x_{min}}, \frac{H * (y - y_{min})}{y_{max} - y_{min}} \right) \quad (1)$$

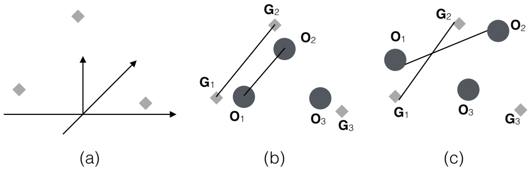

Figure 4: Mapping stroke hold positions from gesture space to scene coordinate system. (a) shows the gestures in the gesture space (Leap sensor coordinate system). In (b), gestures are mapped to the target scene where objects O_1 and O_2 fit with gestures G_1 and G_2 respectively. In (c), gestures are mapped to a distractor scene, where G_1 and G_2 can not fit to objects O_1 and O_2 as well as in the target scene in (b).

In this task, we assume each deictic gesture is meant to refer to one object in the scene. The closer a mapped gesture to an object, the more likely that the object is the referent of the gesture. We trained a Gaussian kernel density estimation (KDE) model f to turn the distance between an object O_i and the gesture G into a probability which indicates how likely the stroke hold fits the object:

$$p(O_i|G) = f(||\mathbf{G} - \mathbf{O}_i||) \quad (2)$$

Details of training and evaluating of the model are described in the evaluation section.

Spatial configuration Individual gestures show how well each gesture fit with a candidate referent. With more than one gestures, the spatial *configuration* of the sequence of gestures can show how well the spatial configuration of the gestures fit with the spatial configuration of objects. As shown in Figure 4(b) and 4(c), the better the spatial configurations of gestures and objects fit with each other, the smaller the angle between the two vectors.

Given two gestures, we estimate the most likely referents with the KDE model and measure how well the gestures fit with the referents by cosine similarity. With $n\,(n \geq 2)$ gestures in a scene description, the probability can be computed as follows:

$$p(O_1, \cdots, O_n|G_1, \cdots, G_n) = \prod_{i=2}^{n} \prod_{j=1}^{i-1} \frac{(\mathbf{G}_i - \mathbf{G}_{i-j}) \cdot (\mathbf{O}_i - \mathbf{O}_{i-j})}{||\mathbf{G}_i - \mathbf{G}_{i-j}||\,||\mathbf{O}_i - \mathbf{O}_{i-j}||} \quad (3)$$

If there is only one gesture, no spatial configuration information is conveyed, therefore the probability is 1. In this way, we incrementally apply gesture information to evaluate how well the gestures fit with a scene.

Language processing: Utterance segmentation

Given word sequences, the segmenter identifies words that start a new description, labels previous word with SEG and informs the NLU module.

We model the segmenter as a sequential classification task, as a new description only starts after previous description is over. Hence, the classifier must therefore learn over a sequence of words to predict segment boundaries. We trained an LSTM network for the task (also using Keras (Chollet 2015)). The network was fed the current word using a one-hot encoding (vocabulary size 266), and had one hidden layer and a sigmoid dense layer that gives the prediction. There are 100 nodes in the hidden layer.

A dropout layer was applied to it to enable more effective learning. 30% of the output units from the hidden layer were randomly selected and set to 0 to avoid overfitting. The loss function of the model was a binary cross entropy loss function. It was optimised with a rmsprop optimiser. The training was stopped when validation loss stopped decreasing.

Note that, the descriptions in the corpus are limited to simple scenes where each description includes three object descriptions, which usually end with object names such as "kreis" (*circle*) and "quadrat" (*square*). Hence, the words are predictive of segmentation boundaries. However, due to the variability of natural descriptions, they also occur in the middle or at the beginning of a segment. For example, segments like "*a red circle here*" or "*circle, on the left, red*" also occur in our corpus. We adopted a simple rule-based segmenter which simply segments object descriptions based on the keywords "*kreis*" and "*quardrat*" as our baseline model, and compared it with the LSTM model (see **Evaluation**).

Language processing: NLU

Given words from a segment U, the NLU module applies a *simple incremental update model* (SIUM) (Kennington and Schlangen 2017; Kennington, Kousidis, and Schlangen 2013) and outputs a probability distribution over all objects in each candidate scene to the *fusion module*. We opted for SIUM for it learns a grounded mapping between words and visual objects and can be applied incrementally:

$$p(O|U) = \frac{1}{p(U)} p(O) \sum_{r \in R} p(U|r) p(r|O) \tag{4}$$

$p(O|U)$ indicates how likely the segment U refers to the object O. The latent variable R takes an object property r from a property set (i.e., *colour*, *shape*, *size*, *vertical* and *horizontal positions*), represented as symbols in the dataset.

$p(U|r)$ is the probability that object property r is described by segment U. It can be learned from data by observing references to objects with that property r. $p(r|O)$ is a normalised distribution over all actual properties of object O. U is represented by ngrams. During application, we marginalised over the properties R of object O, to yield a distribution over candidate objects in a scene. With each word increment, we update $p(O|U)$ by taking the previous distribution as prior for the current step. We then combine $p(O|U)$ with gesture, which will now be explained.

Multimodal fusion & application

The fusion module combines the probability distributions from the two pipelines and retrieves the scene with highest probability. (In the usual terminology, e.g. (Atrey et al. 2010), this is hence a late fusion approach.) Different from a normal late fusion approach, the fusion module in this work includes two steps: 1) multimodal fusion for reference resolution (object level), 2) multimodal fusion of spatial configuration for each scene.

Reference resolution For each segment U, we combine the speech and gesture probability distributions to compute a final probability as follows:

$$p(O_i|U,G) = \lambda_1 * p(O_i|U) + (1-\lambda_1) * p(O_i|G) \tag{5}$$

λ_1 is a weight parameter. When there are no gestures aligned with the segment U, $P(O_i|G)$ is set to 0; λ_1 is set to 1.

Since utterances are segmented into individual object descriptions, we assume each segment U only refers to one object. The object with highest probability is taken as the estimated referent for the description (U, G):

$$O_i^* = \underset{O_i}{argmax}\ p(O_i|U,G) \tag{6}$$

Scene description understanding For each candidate scene C (6 candidate scenes in total, see **Evaluation** for details), we computed a final score by combining the spatial configuration score with the score from previous steps:

$$\begin{aligned} p(C) =& \lambda_2 * \sum_{i=1}^{n} p(O_i)^* \\ &+ (1-\lambda_2) * p(O_1, \cdots, O_n|G_1, \cdots, G_n) \end{aligned} \tag{7}$$

the weight parameter λ_2 determines how much speech contributes to the final decision.

Evaluation

We evaluated individual system components and the whole system with a "hold-one-out" setup. Each time, data from one participant was left as test data while other data as training data to prevent the system from learning about possible idiosyncrasies of a speaker on whom it is tested.

Gesture detector evaluation

The gesture detector classifier achieved an **F1-score** of 0.85, **precision** 0.77, **recall** 0.94. The classification for each stroke hold takes around 10 to 20 ms, correlated to the computational ability of the machine.

Currently, we haven't compared our gesture detector model with other models. Since the main focus of this paper is interpretation and application of the multimodal descriptions, we leave it as future work to implement other models and compare the performance with the current model.

Gesture interpretation evaluation

We evaluated the KDE model by object reference accuracy. Namely, given a gesture position, how often does the referential object get the highest score?

We fit a Gaussian KDE model (with the bandwidth setting to 5) using the distances between mapped gesture positions and referential object positions in the training data.

To test the model, for each gesture in the test set, we computed the distance between the mapped gesture position and all objects in a scene. The model achieves an **average accuracy** of 0.81, which significantly outperforms the chance level baseline 1/3.

Utterance segmenter evaluation

Model	F1-score	Recall	Accuracy	Std of Acc.
`Baseline`	0.84	0.83	0.85	0.23
`LSTM`	0.89	0.88	0.92	0.10

Table 1: Evaluation results of utterance segmenter.

We compared our LSTM model with a rule-based baseline model. As shown in Table 1, the LSTM model outperforms the baseline model. It achieves a higher accuracy score and the standard deviation of accuracy is lower which indicates stable performances between participants. This is consistent with our observation that the verbal descriptions are natural and there are individual differences between participants even within the task constraints. It's likely that with more training data, the LSTM model will perform even better.

The NLU component is evaluated as part of the whole system evaluation show in the next section.

	Speech only	Gesture only	Multimodal
Human eval	0.86 / -	0.32 / -	0.77 / -
Our System	0.75 / 0.79	0.50 / 0.50	0.84 / 0.85
Baseline	0.17 / 0.41	0.17 / 0.41	0.17 / 0.41

Table 2: Evaluation results (accuracy / MRR). See text for *human eval* method.

Whole system evaluation

To assess the whole system, we designed offline tests in unimodal and multimodal setups. We simulated the multimodal interaction by playing back multimodal descriptions in realtime. The transcriptions of speech were played back, simulating output of an incremental ASR, and stroke hold positions detected from the motion data were played back as gesture positions.

We created a test set for each scene in the corpus. Each test set includes the target scene and 5 randomly selected distractor scenes. Hence, the chance level accuracy of the scene retrieving task is 0.17.

The metrics We evaluated the system performance with accuracy and **mean reciprocal rank (MRR)**:

$$\mathit{MRR} = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \tag{8}$$

$|Q|$ indicates the set of scene retrieval queries. For each scene retrieval, we rank the candidate scenes according to the scores from the fusion module (e.g., the scene with the highest score got a rank of 1). $rank_i$ ranges from 1 to 6. **MRR** ranges from 0.41 (the worst case) to 1 (the ideal case).

Speech only In this test, only speech contributes information to the decision. I.e., λ_1 in Equation 5 and λ_2 in Equation 7 equal 1.

As shown in Table 2, the average **MRR** of the tests is 0.79. It significantly outperforms the baseline. Comparing evaluation score of each participants, we observed individual differences between speakers. Given that the evaluation setups are the same for all the participants, the difference could be due to varied language descriptions (e.g., omitting spatial relations in language descriptions and only referring to objects with visual properties), which affects the performance of the utterance segmenter and the NLU model.

Gesture only In this test, we set the weight of language descriptions (i.e., λ_1 in equation 5 and λ_2 in 7) in the fusion module to 0, so that only gestures contribute to making the scene retrieving decision.

The average accuracy of all tests is 0.50. It outperforms the baseline while underperforms the multimodal model. Note that gestures only convey positional information and the spatial configuration of the referents, the similarity between targeted scene and distractors also affect the results.

Speech + gesture In this test, the fusion module combined information from speech and gestures. We assumed that speech and gestures contribute equally, therefore, λ_1 in equation 5 and λ_2 in 7 were set to 0.5. The average **MRR** of all tests is 0.84. It shows that gestures help to improve the system performance.

Discussion The results show that our method can successfully extract spatial information from gestures. The "speech-only" condition also achieves good performance. Combining speech and gestures further improves the system performance, although the improvement is somewhat limited. One reason for this is that position information is often redundantly encoded in verbal descriptions. Overlap in content between gestures and speech has been observed in previous works (Epps, Oviatt, and Chen 2004); the data collection setup may have further encouraged such redundant encoding. In real situations, it may be less likely that speakers indeed mention all attributes, in which case contributions of modalities may be more complementary. (The system, in any case, would be ready to handle this.) In a practical system, this redundancy might even be a useful feature. Here, we allowed the system to incrementally access the manual transcription of the speech. In a live system, verbal descriptions would come from automatic speech recognition (ASR), and would be more noisy. The redundancy coming from the gestures will then help locally disambiguate the ASR output. We will test this in future work.

To ground our results in human performance, we randomly selected 65 scene descriptions from the corpus and asked crowd workers from the Crowdflower platform to perform the scene retrieving task. Using audio alone, the participants beat our model (see Table 2); in the video-alone condition, our system performs better. Interestingly, the full condition (language + gestures) actually seems to be distracting to the participants, compared to language-only. This is presumably due to the cognitive load of observing the gestures and evaluating the scenes. We take this as support for the assumption that in a real interaction, the delivery of such descriptions would be much more interactive and delivered in installments. To model this, a system needs to interpret incrementally. We evaluate our incremental performance next.

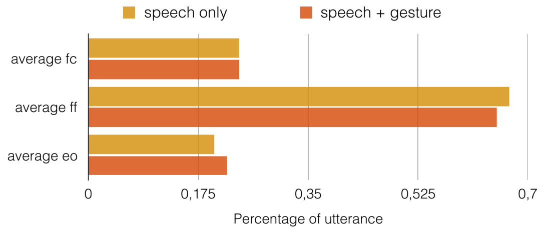

Figure 5: Results of incremental evaluation. See text for description of metrics. Lower is better.

Incremental evaluation

We evaluated the system performance in speech only and speech plus gesture setups on the incremental level, using incremental evaluation metrics (Baumann, Buß, and Schlangen 2011; Buß and Schlangen 2010):

- **average fc (first correct):** how deep into the utterance (as percentage of the whole utterance duration) does the system makes a correct decision the first time, potentially changing its mind again later?
- **average ff (first final):** how deep into the utterance does the system makes a correct final guess?
- **average eo (edit overhead):** ratio of necessary edits/all edits, indicating how stable the system's decisions are.

As shown in Figure 5, in both cases, the average **fc** is 0.24. Gestures cannot help to make the first correct decision earlier. It is because with only one gesture, the gesture doesn't contribute spatial configuration information to differentiate between candidates. Speakers often start a description with speech, therefore speech comes earlier than gestures and contributes earlier than gestures.

However, gestures do help to make the first final decision slightly earlier, with an average **ff** of 0.65, comparing with a value of 0.67 in speech only situation. For example, for an utterance of 30 s, the precedence translates to 600 ms, which is noticeable. In our corpus, descriptions usually end with speech. Gestures complete earlier than speech. Given all gestures in a description, the spatial configuration in gestures contributes a lot to identify the targeted scene since all scene candidates have different spatial configurations.

When combining speech with gestures, the average **eo** is slightly higher. It shows that gestures complement speech in the task. With the information gestures contributed, the system risks more edits to move toward a right decision. The reward of more edits is an earlier first final correct decision.

Figure 6 plots **MRR** over the course of the utterances (to be able to average, again expressed as percentage of full utterance). **MRR** increases continuously, indicating that for this task, important information can still come late.

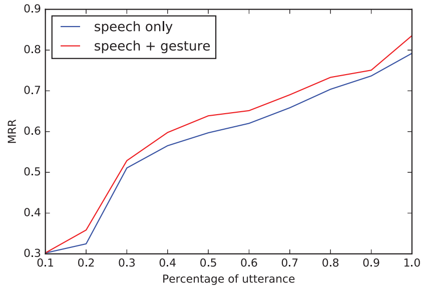

Figure 6: Average MRR of incremental evaluation.

Discussion On the incremental level, gestures help to achieve an earlier correct final decision. It's promising that in situated dialogues, the system might understand descriptions from humans without waiting for all verbal descriptions, thus may behave more human-like. Moreover, gestures result in more overhead edits (Figure 6). This signal can be used for clarifications in situated dialogues. For instance, while a route giver notices that the system's decision changes to bad decisions, the route giver might change the description strategy to make the decoding task easier, or the system can make clarification requests. These signals will lead to more human-like interactions.

Conclusion and future work

In this paper, we modelled the interpretation of multimodal spatial descriptions, a common scenario in route giving tasks. The evaluation results in uni-modal setups show that both speech and gestures are informative. Combining speech and gestures further improved the system performance. Furthermore, we evaluated the system incrementally. The results show that gestures help to achieve earlier final correct decisions. Hence, gestures not only contribute information, but also benefit interpretations on the incremental level due to its parallel nature with speech.

The language grounding model (SIUM) has been shown to work in larger domains. We believe that our system forms a good basis for scaling this up to online route description understanding, for example for mobile robots. This will add the orthogonal challenges of resolving descriptions of actual routes within the identified map areas, and resolving more complex landmark descriptions, which may well contain iconic gestures to describe building shapes (Cassell et al. 2007). An open question is how more complex constructions, for example involving quantifiers or negation, should be modelled. In the future, we will test our system in realistic human-robot interaction scenarios with more general interpretation space.

Acknowledgments

This work was supported by China Scholarship Council and the Cluster of Excellence Cognitive Interaction Technology CITEC (EXC 277) at Bielefeld University, which is funded by the German Research Foundation (DFG).

References

Alibali, M. 2005. Gesture in spatial cognition: Expressing, communicating, and thinking about spatial information. *Spatial Cognition and Computation* 5(4):307–331.

Atrey, P. K.; Hossain, M. A.; El Saddik, A.; and Kankanhalli, M. S. 2010. Multimodal fusion for multimedia analysis: a survey. *Multimedia systems* 16(6):345–379.

Baumann, T., and Schlangen, D. 2012. The inprotk 2012 release. In *NAACL-HLT Workshop on Future Directions and Needs in the Spoken Dialog Community: Tools and Data*, 29–32. Association for Computational Linguistics.

Baumann, T.; Buß, O.; and Schlangen, D. 2011. Evaluation and Optimization of Incremental Processors. *Dialogue and Discourse* 2(1):113–141.

Boularias, A.; Duvallet, F.; Oh, J.; and Stentz, A. 2016. Learning qualitative spatial relations for robotic navigation. In *IJCAI*, 4130–4134.

Buß, O., and Schlangen, D. 2010. Modelling sub-utterance phenomena in spoken dialogue systems. In *Proceedings of the 14th International Workshop on the Semantics and Pragmatics of Dialogue (Pozdial 2010)*, 33–41.

Campana, E.; Silverman, L.; Tanenhaus, M. K.; Bennetto, L.; and Packard, S. 2005. Real-time integration of gesture and speech during reference resolution. In *Proceedings of the 27th annual meeting of the Cognitive Science Society*, 378–383. Citeseer.

Cassell, J.; Kopp, S.; Tepper, P.; Ferriman, K.; and Striegnitz, K. 2007. Trading spaces: How humans and humanoids use speech and gesture to give directions. *Conversational informatics* 133–160.

Chollet, F. 2015. Keras. https://github.com/fchollet/keras.

Duvallet, F.; Walter, M. R.; Howard, T.; Hemachandra, S.; Oh, J.; Teller, S.; Roy, N.; and Stentz, A. 2016. Inferring maps and behaviors from natural language instructions. In *Experimental Robotics*, 373–388. Springer.

Duvallet, F.; Kollar, T.; and Stentz, A. 2013. Imitation learning for natural language direction following through unknown environments. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, 1047–1053. IEEE.

Emmorey, K.; Tversky, B.; and Taylor, H. a. 2000. Using space to describe space: Perspective in speech, sign, and gesture. *Spatial Cognition and Computation* 2(3):157–180.

Epps, J.; Oviatt, S.; and Chen, F. 2004. Integration of speech and gesture inputs during multimodal interaction. In *Proc Aust. Int. Conf. on CHI*.

Hemachandra, S.; Walter, M. R.; Tellex, S.; and Teller, S. 2014. Learning spatial-semantic representations from natural language descriptions and scene classifications. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, 2623–2630. IEEE.

Kendon, A. 1980. Gesticulation and speech: two aspects of the process of utterance. *The Relationship of Verbal and Nonverbal Communication* 25:207–227.

Kennington, C., and Schlangen, D. 2017. A simple generative model of incremental reference resolution for situated dialogue. *Computer Speech & Language* 41:43–67.

Kennington, C.; Kousidis, S.; and Schlangen, D. 2013. Interpreting Situated Dialogue Utterances: an Update Model that Uses Speech, Gaze, and Gesture Information. In *SIGdial 2013*.

Kollar, T.; Tellex, S.; Roy, D.; and Roy, N. 2010. Toward understanding natural language directions. In *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*, 259–266. IEEE.

Kousidis, S.; Pfeiffer, T.; and Schlangen, D. 2013. MINT . tools : Tools and Adaptors Supporting Acquisition , Annotation and Analysis of Multimodal Corpora. In *Proceedings of Interspeech 2013*, 2649–2653. Lyon, France: ISCA.

Matuszek, C.; Bo, L.; Zettlemoyer, L.; and Fox, D. 2014. Learning from unscripted deictic gesture and language for human-robot interactions. In *AAAI*, 2556–2563.

McNeill, D. 1992. Hand and Mind: What Gestures Reveal About Thought. *What gestures reveal about* 1–15.

Mei, H.; Bansal, M.; and Walter, M. R. 2016. Listen, attend, and walk: Neural mapping of navigational instructions to action sequences. In *AAAI*, 2772–2778.

Roy, D. K. 2002. Learning visually grounded words and syntax for a scene description task. *Computer speech & language* 16(3):353–385.

Schneider, L. F., and Taylor, H. a. 1999. How do you get there from here? Mental representations of route descriptions. *Applied Cognitive Psychology* 13(September 1998):415–441.

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

Stiefelhagen, R.; Fugen, C.; Gieselmann, R.; Holzapfel, H.; Nickel, K.; and Waibel, A. 2004. Natural human-robot interaction using speech, head pose and gestures. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, volume 3, 2422–2427.

Tellex, S.; Kollar, T.; Dickerson, S.; Walter, M. R.; Banerjee, A. G.; Teller, S. J.; and Roy, N. 2011. Understanding natural language commands for robotic navigation and mobile manipulation. In *AAAI*.

Whitney, D.; Eldon, M.; Oberlin, J.; and Tellex, S. 2016. Interpreting multimodal referring expressions in real time. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, 3331–3338. IEEE.

Williams, T.; Schreitter, S.; Acharya, S.; and Scheutz, M. 2015. Towards situated open world reference resolution. In *AAAI Fall Symposium on AI for HRI*.

Williams, T.; Acharya, S.; Schreitter, S.; and Scheutz, M. 2016. Situated open world reference resolution for human-robot dialogue. In *The Eleventh ACM/IEEE International Conference on Human Robot Interaction*, 311–318. IEEE Press.