Communicating and Acting: Understanding Gesture in Simulation Semantics

Nikhil Krishnaswamy[1], Pradyumna Narayana[2], Isaac Wang[3], Kyeongmin Rim[1], Rahul Bangar[2], Dhruva Patil[2], Gururaj Mulay[2], Ross Beveridge[2], Jaime Ruiz[3], Bruce Draper[2], and James Pustejovsky[1]

[1]Dept. of Comp. Sci., Brandeis University, Waltham, MA, USA
[2]Dept. of Comp. Sci., Colorado State University, Fort Collins, CO, USA
[3]Dept. of Comp. and Info. Sci. and Eng., University of Florida, Gainesville, FL, USA
{nkrishna,krim,jamesp}@brandeis.edu, {prady,rahul.bangar,dkpatil,guru5, ross.beveridge,draper}@colostate.edu, {wangi,jaime.ruiz}@ufl.edu

Abstract

In this paper, we introduce an architecture for multimodal communication between humans and computers engaged in a shared task. We describe a representative dialogue between an artificial agent and a human that will be demonstrated live during the presentation. This assumes a multimodal environment and semantics for facilitating communication and interaction with a computational agent. To this end, we have created an embodied 3D simulation environment enabling both the generation and interpretation of multiple modalities, including: language, gesture, and the visualization of objects moving and agents performing actions. Objects are encoded with rich semantic typing and action affordances, while actions themselves are encoded as multimodal expressions (programs), allowing for contextually salient inferences and decisions in the environment.

1 Introduction

In order to facilitate collaborative communication between a human and a computational agent, we have been working to integrate a multimodal model of semantics (*Multimodal Semantic Simulations, MSS)* with a real-time visual recognition system for identifying human gestures. The language VoxML, Visual Object Concept Modeling Language (Pustejovsky and Krishnaswamy, 2016), is used as the platform for multimodal semantic simulations in the context of human-computer communication. Gestural input is recognized in real time by a convolutional neural net-based machine vision system networked to the simulation environment, which is configured for joint activity and communication between a human and a computational agent. This involves the integration of inputs from speech, gesture, and action, as mediated through a dialogue manager (DM) that tracks discourse and situational context variables embodied in a shared situated simulation. Hence, the dynamic of human-computer interaction changes from giving and receiving orders to a *peer-to-peer conversation*.

We explore this idea in the context of the blocks world. In particular, we consider a scenario in which one person (the *builder*) has a table with blocks that another person (the *signaler*) can see. We also assume the builder and signaler can see each other. The signaler is then given a pattern of blocks, and their job is to get the builder to recreate the pattern. While blocks world is obviously not a real-world application, it serves as a surrogate for any cooperative task where both partners share a workspace.

Our system design and gesture vocabulary are taken primarily from an elicitation study, similar to that introduced in Wobbrock et al. (2009) but with differences in the way gestures are elicited. The purpose of the study underlying our gesture vocabulary (Wang et al., 2017a,b) was to analyze the natural dyadic communication used by two people when engaging in solving a collaborative task.

We asked pairs of participants to collaboratively build different pre-determined structures using wooden blocks. Participants were put in separate rooms with similar setups. Each participant stood in front of a table facing a TV screen on the opposite end of the table. Microsoft Kinect v2 sensors were

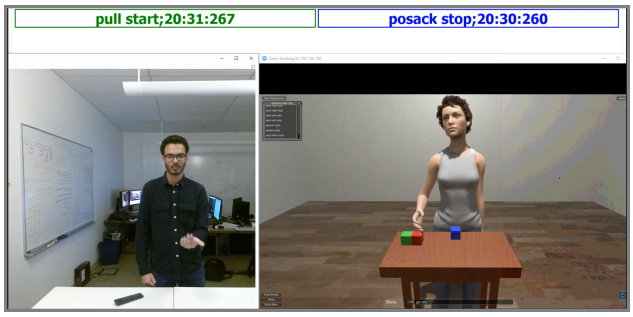

Figure 1: Prototype peer-to-peer interface

also set up on the opposite end, facing the participant. We developed software to stream live video (and audio) from the Kinect sensors between the two setups so that participants could communicate with each other as if they were facing each other at opposite ends of the same table. The Kinect sensors were also used to record the experiment, providing us with RGB video, depth data, and motion capture skeletons.

One participant was given the role of builder and was provided with a set of 12 wooden cubes (with 4-inch sides). The other participant was given the role of signaler and was given an image of an arrangement, or layout, of these blocks. The signaler was assigned the task of communicating to and directing the builder to replicate the layout; the builder needed to respond to the signalers commands by placing and arranging the blocks on the table. The table acted as a shared workspace, as blocks placed on the table could be seen by both participants (although from opposite perspectives). Not all 12 blocks were used for every layout, and the signaler was not allowed to show the layout to the builder.

Since we wanted to observe natural communication in action, participants were also not allowed to talk or strategize beforehand, and no instruction on how to speak/gesture was given from the experimenter. A trial began when the experimenter presented a new block layout to the signaler and ended when the participants replicated the block layout. Communication between the two varied across three conditions:

(1) the signaler and builder could both see and hear each other;

(2) the signaler and builder could see but not hear each other;

(3) the signaler could see the builder (and therefore the blocks on the table), but the builder can only hear the signaler.

In our working prototype human-computer system, the signaler is a person and the builder is an avatar, with a virtual table and virtual blocks. The signaler can see a graphical projection of the virtual world, and communicates to the avatar through gestures. The avatar communicates back through language (text or speech), gesture, and action in the form of moving blocks (cf. Figure 1). In particular, we took initial inspiration for the system design from the setup within the elicitation study where the participants could only see but not hear each other, removing the audio channel entirely and forcing them to rely on gestures to communicate, in order to assess the impact that gesture had on the communication. Therefore the initial setup only allowed the signaler to communicate though gesture, and subsequent refinements have introduced speech and language input to the signaler's capability, increasing the level of communicative symmetry in the interaction.

2 Related Work

Multimodal interfaces combining language and gesture are found in the literature since Bolt's "Put-that-there" system (1980), which anticipated some of the issues discussed herein, including the use of deixis

to disambiguate references, and also inspired a community surrounding multimodal integration (e.g., Dumas et al. (2009); Kennington et al. (2013); Turk (2014)).

The psychological motivation for multimodal interfaces, as epitomized by Quek et al. (2002), holds that speech and gesture are coexpressive and processed partially independently, and therefore complement each other. Using both modalities increases human working memory and decreases cognitive load (Dumas et al., 2009), allowing people to retain more information and learn faster.

Visual information has been shown to be particularly useful in establishing common ground (Clark and Wilkes-Gibbs, 1986; Clark and Brennan, 1991; Dillenbourg and Traum, 2006; Eisenstein et al., 2008b,a), or mutual understanding that enables further communication. Other research in HCI additionally emphasizes the importance of shared visual workspaces in computer-mediated communication (Fussell et al., 2000, 2004; Kraut et al., 2003; Gergle et al., 2004), highlighting the usefulness of non-verbal communication in coordination between humans (Cassell et al., 2000; Cassell, 2000).

Brennan et al. (2008) shows that allowing for shared gaze increased performance in spatial tasks in paired collaborations. Multimodal systems of gaze and speech have also been studied in interaction with robots and virtual avatars (Andrist et al., 2017; Mehlmann et al., 2014; Skantze et al., 2014). However, few systems have centered the use of language and gesture in collaborative and communicative scenarios.

Communicating with computers becomes even more interesting in the context of shared physical tasks. When people work together, their conversation consists of more than just words. They gesture and they share a common workspace (Lascarides and Stone, 2006, 2009b; Clair et al., 2010; Matuszek et al., 2014). Their shared perception of this workspace is the context for their conversation, and it is this shared space that gives many gestures, such as pointing, their meaning (Krishnaswamy and Pustejovsky, 2016a). The dynamic computation of discourse (Asher and Lascarides, 2003), furthermore, becomes more complex when multiple modalities are at play. Fortunately, embodied actions (such as coverbal gestures) do not seem to violate coherence relations (Lascarides and Stone, 2009a).

Many of the components used here will be familiar, in role if not in details. Visual gesture recognition has long been a challenge (Jaimes and Sebe, 2007; Madeo et al., 2016). Gesture recognition in this system is facilitated by Microsoft Kinect depth sensing (Zhang, 2012) and ResNet-style deep convolutional neural networks (DCNNs) (He et al., 2016) implemented in TensorFlow (Abadi et al., 2016).

The avatar and her virtual blocks world are implemented with VoxSim, a semantically-informed reasoning system previously described by Krishnaswamy and Pustejovsky (2016b) that allows the avatar to react to gestural events with both actions and words.

3 Communicating through Gesture, Language and Action

The system operates in real time, allowing the human signaler to gesture to the avatar. In return, the avatar can gesture, speak with the words also printed on screen, or communicate through actions by moving blocks. This system, the human/avatar blocks world (HAB) allows us to explore peer-to-peer communication between people and computers.

While the HAB implementation relies on many components, here we focus on the real-time gesture recognition module, which recognizes gestures by the signaler, the grounded semantics module (VoxSim), which determines the avatar's response to gestures, and the interplay between them. VoxSim is described in Subsection 3.1, gesture recognition is described in Subsection 3.2, while the interactions between the two are described in Subsection 3.3.

3.1 VoxSim

The HAB system's virtual world is built on the VoxSim platform (Krishnaswamy and Pustejovsky, 2016a,b), an open-source, semantically-informed 3D visual event simulator implemented in the Unity game engine (Goldstone, 2009) that leverages game engine graphics processing, UI, and physics to operationalize events described in natural language within a virtual environment.

VoxSim maps natural language event semantics through a dynamic interval temporal logic (DITL) (Pustejovsky and Moszkowicz, 2011) and the modeling language VoxML (Pustejovsky and Krishnaswamy, 2016). VoxML encodes qualitative and geometrical knowledge about objects and events that is presupposed in linguistic utterances but not made explicit, in a visual modality. This includes information about symmetry or concavity in an object's geometry, the relations resulting from an event, the qualitative relations described by a positional adjunct, or behaviors *afforded* by an object's *habitat* (Pustejovsky, 2013; McDonald and Pustejovsky, 2014) associated with the situational context that enables or disables certain actions that may be undertaken using the object. Such information is a natural extension of the lexical semantic typing provided within Generative Lexicon Theory (Pustejovsky, 1995), cf. also Asher (2011), towards a semantics of embodiment. This allows the HAB system to determine which regions, objects, or parts of objects may be indicated by gestures such as deixis or action referentials, and the natural language interface allows for human-understandable disambiguation. Object motion and agent motion are compositional in the VoxML framework, allowing VoxSim to easily separate them in the virtual world, so the gesture used to refer to an action (or program) can be directly mapped to the action itself, establishing a shared context grounded from the perspective of both the human and the computer program.

3.1.1 Dialogue Manager

Avatar-directed dialogue serves to manage the flow of control through either requesting disambiguation in the previously-established context, acknowledging receipt of a gesture, or expressing completion of an action. Dialogue output from the avatar is usually accompanied by a complementary gesture, such as deixis of a block or region, or enactment of a program over an object. The dialogue manager (DM) maintains a queue of possible outputs based on what additional information the avatar needs to know to complete its next action, including unique disambiguating attributes of the blocks (e.g., distinguishing color) or disambiguating labels of the actions (e.g., possible relational interpretations of a gesture, relative to the table or a block). It then composes questions or statements based on these qualities, in order to give the human signaler the most complete amount of information needed to interpret the received gestures.

For instance, when presented with a gesture indicating a region of the table that currently contains a green block and a red block, the dialogue manager takes the possible entities indicated (here, the set of two blocks), and calculates those attributes unique to each one (here, distinct colors). The avatar can then present the human with questions of the form "*Are you pointing to the* [COLOR] *block?*" that the human can answer either in the affirmative or negative, in order to communicate their intent. This same process can be used to disambiguate the particulars of actions, such as requesting clarification about the location to which a block is intended to be moved. Some specific examples of this are discussed in Sections 3.3 and 3.4.

While having the avatar explicitly ask if the human is indicating a particular block or action is one way of replicating a naturalistic interaction, having this repetitive process be the only way of resolving ambiguity does not exercise the variety of methods humans would use with each other in the same task. Particularly, if many options must be iterated through before the avatar arrives at the intended one, then the interaction risks becoming tedious. To this end, it may often be more useful and more naturalistic for the avatar to ask an open-ended question that the human can answer using one of their available modalities. For example when presented with an ambiguous block choice, the avatar might instead ask the human "Which block?" to which the human can respond with some distinguishing attribute as has already been determined for the possible blocks under question, such as relative orientation or color. Something like relative orientation can be easily communicated through gesture, as in the context of the question very coarse-grained directional deixis can suffice to distinguish the "left" block from the "right" block or the "near you" block from the "near me" block. An attribute like color cannot be easily communicated through gesture, but we have been working on integrating speech input on the human's side, and so a question such as "Which block?" could be answered with "The red one" or simply "red," and VoxSim can map that attribute to the block in the scene that has it. The addition of speech recognition to the human's input, and speech synthesis to the avatar's output parallels the gesture recognition capability and the animated avatar gesture generation in VoxSim, bringing a measure of

symmetry to the communication in both modalities.

3.2 Gesture Recognition

The gesture recognition module independently labels five body parts. The left and right hands are both labeled according to their pose. The system is trained to recognize 34 distinct hand gestures in depth images, plus an "other" label, used for hands at rest or in unknown poses. Hand poses are directional, such that pointing down is considered a different pose than pointing to the right. Head motions are classified as either nod, shake or other based on a time window of depth difference images. Finally, the left and right arms are labeled according to their direction of motion, based on the skeleton pose estimates generated by the Microsoft Kinect (Zhang, 2012).

Real-time gesture recognition is spread across 6 processors, as shown in Figure 2: *Kinect Host* segments hands and head data from skeleton data gathered from the Kinect atop the signaler's monitor, producing 3 depth streams; *Right Hand Pose*, *Left Hand Pose*, and *Head Motion* are each ResNet-style deep convolutional neural networks (DCNNs) (He et al., 2016) on nVidia Titan X GPUs; *Arm Motion* labels arm directions from skeleton data; *Gesture Fusion* collects the hand, arm and head labels and fuses them using finite state machines to detect gestures.

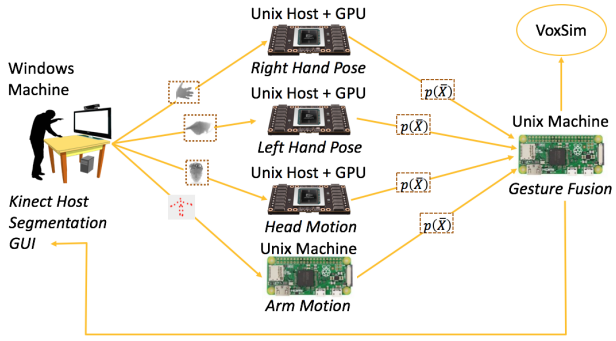

Figure 2: The architecture of the real-time gesture recognition module

3.3 Gestures & VoxSim

VoxSim receives gestural "words" from the gesture recognizer, and interprets them at a contextually-wrapped compositional semantic level. For the moment, interface is limited to seven such "words" chosen due to their frequent occurrence in the elicitation study described in Section 1:

1. Engage. Begins the task when the signaler steps up to the table, and ends it when they step back.
2. Positive acknowledge. A head nod or a "thumbs up" pose with either or both hands. Used to signal agreement with a choice by the avatar or affirmative response to a question.
3. Negative acknowledge. A head shake, "thumbs down" with either or both hands, or palm-forward "stop" sign. Signals disagreement with a choice by the avatar or negative response to a question.
4. Point. Pointing (deixis) includes the direction of the hand and/or arm motion: one or two of front, back, left, right, up, or down. Indicates a region or block(s) in that region.
5. Grab. A "claw," mimicking grabbing a block. Tells the avatar to grasp an indicated block.
6. Carry. Moving the arm in a direction while the hand is in the grab position. "Carry up" can be thought of as *pick up*, while "carry down" is equivalent to *put down*.
7. Push. A flat hand moving in the direction of the open palm. Like "carry," but without the up or down directions. A beckoning motion signals the avatar to push a block toward the signaler.

Importantly, the semantics of each of these gestures is *compositional* and *context-dependent*. The "directional" gestures, that is "point," "carry," and "push," require a direction for a complete interpretation. Pointing provides the indexicals or "nouns" that are used as arguments for subsequent action gestures or "verbs."

Context is typically introduced through objects or possible objects indicated. For example, "grab" has a distinct interpretation in the blocks world context, but could be interpreted and enacted differently over a different set of objects, particularly if those objects afford grasping in a manner different from the claw-like hand position used with small blocks.

All gestures are interpreted in the current context, which is established by previously undertaken actions by both the human and the avatar. For instance, if at the beginning of the scene, the human first makes the "grab" gesture as described above, and has not indicated a block which they intend the avatar to grasp, the gesture is not provided with enough context to be completely interpreted and the avatar must return with a question of uncertainty; she cannot, from the information provided by the gesture and current context, determine what is meant. If the human begins by pointing to a region of the table, the context makes it necessary to look for potential blocks that must be indicated. If no blocks exist in that area of the table the avatar must say she doesn't know what the human means to indicate; if more than one block exists there, she must ask for clarification. Examples of these ambiguities are discussed below and in Section 3.4.

The avatar can communicate its interpretation and intent to the human in multiple modalities. The human can see the avatar enacting a command in context when it moves blocks in the virtual world. The avatar can also communicate through gesture, for example by reaching toward a block. Finally, the avatar can speak to the signaler through text or speech output, to initiate requests for clarification if necessary.

When VoxSim receives a semantic gesture from the recognition module, it parses the meaning of the gesture in the context of the current state occupied by the avatar and the blocks. For example, if the humans points to the right and the avatar is currently holding a block, then the gesture may be a request to move the block to the right. Alternatively, if the avatar is not currently holding a block, the same gesture may indicate a block on the right side of the table for the next action.

Gestural ambiguities are common. If there are two blocks on the right side of the table and the signaler points to the right, which block do they mean? Similarly, a gesture to put a block down may be ambiguous: should the avatar put the block down on top of the block below it, or next to it?

When presented with ambiguous gestures, VoxSim asks the signaler to choose among the possible interpretations. VoxSim orders the options according to a set of heuristics derived from common human intentions observed in the aforementioned elicitation studies. For example, if the options are to put a red block on top of a blue block or next to it, VoxSim favors the stacking option. It will ask the signaler if the first option is the desired choice (e.g., "Should I put the red block on top of the blue block?"). It will iterate through the options until it receives a positive acknowledgement or runs out of options, in which case it tells the signaler that it does not understand and waits for a new gesture.

3.4 Building a Three-block Staircase

Figure 3 depicts the process of building a three-block staircase, an example scenario as can be demonstrated using the HAB system. In Frame A, the signaler, having engaged the avatar in the task, points to the left, which the avatar interprets as indicating the blue block and gestures to that block, in acknowledgment. In Frame B, having directed the avatar to move the blue back away from her, the signaler points to the right. This is ambiguous, since there are two blocks there, so the avatar asks if the signaler means the red block. The signaler shakes his head, so the avatar asks about the green block. The signaler nods, then directs the avatar to push the green block toward him. The signaler points to the blue block, then gestures to slide it to the right (Frame C). The avatar asks for disambiguation: should it slide the block all the way to the green block? The signaler gives this option a thumbs up (Frame D).

Initiative continues to switch. The signaler indicates the red block and directs the avatar to pick up and move it. Frame E shows him performing a "carry" gesture up and away from himself. He then lowers his arm to indicate putting the block down. Placement is again ambiguous – should the red block

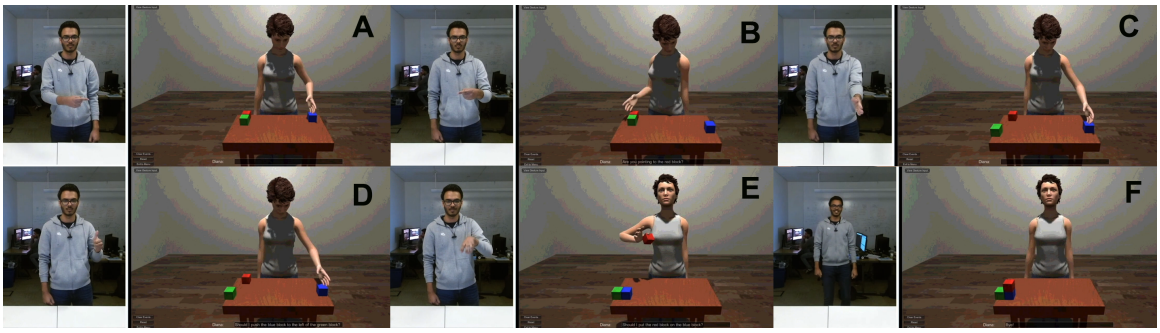

Figure 3: An example of building a staircase in HAB

go on the blue block, the green block, or the table top? The avatar asks and resolves these questions, and Frame F shows the completed staircase.

4 Conclusion and Future Directions

In this paper we have demonstrated the semantics and architecture underlying a multimodal peer-to-peer interface that interprets natural gestures in the context of a shared perceptual task and uses a mixed initiative dialogue. The result is symmetric communication between a human and a virtual avatar on a shared task with a common goal, facilitated by the shared context created through linguistic, gestural, and visual modalities, that highlights certain dynamics of multimodal communication, including:

- Judiciously determining when to have the avatar seek confirmation from the human as opposed to having it confirm every action;
- Gestural communication from the avatar to the human, such as gesturing toward a potentially-indicated block or region to seek confirmation, as doing nothing suggests to the human that no gesture was received by the system;
- A common coordinate system when using deixis as a gestural "referring expression," so that directions like "forward"/"back" or "left"/"right" have a common interpretation to both parties.

The HAB system's current implementation as demonstrated is limited to a vocabulary of well-defined gestures that appeared frequently in the human-subject elicitation studies discussed in Section 1. This demonstration serves to show how very basic actions can be mapped through a dynamic event semantics that interprets both natural language utterances and gestures, to facilitate the completion of a shared task and to communicate a goal from a human to a computer.

Further work to be undertaken primarily involves making interactions enabled through the HAB system more naturalistic. With the addition of speech recognition and generation to the avatar side, we are more closely approaching a true symmetry of communication between the interlocutors, and so in order to increase the variety of instruction and action that can be communicated between them, we need to increase the range of gesture semantics the system can process and compose. We will begin with other common gestures that occurred frequently in the aforementioned elicitation studies. These include the introduction of scalars, such as cardinal numbers (indicated by fingers and indexicals) and spacing (as indicated by space between the hands). This might map to a linguistic utterance such as "place three blocks *this far* apart," where "three" is accompanied by three raised fingers and "this far" is accompanied by separated hands indicating a distance. In the near term, we are also researching the inclusion of iterated actions, i.e., a palm-extended "push" hand continually moving over the same short interval, such as might represent an utterance of "a little more" or "keep going." This allows more fine-grained control over the placement of blocks and speed of motion.

We are also exploring approaches to performing automatic compositions of motions from primitives, including approaches based on machine learning (cf. Dubba et al. (2015); Do and Pustejovsky (2017)) and analogical reasoning (cf. Chen and Forbus (2017)). Such composition-based approaches would

allow the agent to be taught a behavior with a given label, and then replicate it, e.g., following the instructions given in the example scenario in Section 3.4, label the resulting structure a "staircase," and then instruct the agent to build another staircase or "build another one."

Finally, by extending the virtual environment to robotics, wherein a real robotic agent enacts actions in an environment isomorphic to the simulated environment, the simulated environment provides a context within which new concepts can be grounded (cf. Paul et al. (2016)).

This system, and the addition of gestural interpretation to VoxSim demonstrates a viable platform for cooperating with computational agents in a new way, by creating a common ground that the human and machine partners can share for demonstrating individual and common knowledge, action, and planning. Together, they take the steps to complete tasks through mixed initiative conversations.

Acknowledgements

We would like to thank the reviewers for their insightful comments. We have tried to incorporate their suggestions where possible. This work was supported by Contract W911NF-15-C-0238 with the US Defense Advanced Research Projects Agency (DARPA) and the Army Research Office (ARO). Approved for Public Release, Distribution Unlimited. The views expressed herein are ours and do not reflect the official policy or position of the Department of Defense or the U.S. Government. All errors and mistakes are, of course, the responsibilities of the authors.

References

Abadi, M., P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. (2016). Tensorflow: A system for large-scale machine learning. In *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI). Savannah, Georgia, USA*.

Andrist, S., M. Gleicher, and B. Mutlu (2017). Looking Coordinated: Bidirectional Gaze Mechanisms for Collaborative Interaction with Virtual Characters. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, New York, NY, USA, pp. 2571–2582. ACM.

Asher, N. (2011). *Lexical meaning in context: A web of words*. Cambridge University Press.

Asher, N. and A. Lascarides (2003). *Logics of conversation*. Cambridge University Press.

Bolt, R. A. (1980). *"Put-that-there": Voice and gesture at the graphics interface*, Volume 14. ACM.

Brennan, S. E., X. Chen, C. A. Dickinson, M. B. Neider, and G. J. Zelinsky (2008, March). Coordinating cognition: The costs and benefits of shared gaze during collaborative search. *Cognition 106*(3), 1465–1477.

Cassell, J. (2000). *Embodied conversational agents*. MIT press.

Cassell, J., M. Stone, and H. Yan (2000). Coordination and context-dependence in the generation of embodied conversation. In *Proceedings of the first international conference on Natural language generation-Volume 14*, pp. 171–178. Association for Computational Linguistics.

Chen, K. and K. D. Forbus (2017). Action recognition from skeleton data via analogical generalization. In *Proc. 30th International Workshop on Qualitative Reasoning*.

Clair, A. S., R. Mead, M. J. Matarić, et al. (2010). Monitoring and guiding user attention and intention in human-robot interaction. In *ICRA-ICAIR Workshop, Anchorage, AK, USA*, Volume 1025.

Clark, H. H. and S. E. Brennan (1991). Grounding in communication. In L. Resnick, L. B., M. John, S. Teasley, and D (Eds.), *Perspectives on Socially Shared Cognition*, pp. 13–1991. American Psychological Association.

Clark, H. H. and D. Wilkes-Gibbs (1986, February). Referring as a collaborative process. *Cognition 22*(1), 1–39.

Dillenbourg, P. and D. Traum (2006). Sharing solutions: Persistence and grounding in multimodal collaborative problem solving. *The Journal of the Learning Sciences 15*(1), 121–151.

Do, T. and J. Pustejovsky (2017). Learning event representation: As sparse as possible, but not sparser. In *Proc. 30th International Workshop on Qualitative Reasoning*.

Dubba, K. S., A. G. Cohn, D. C. Hogg, M. Bhatt, and F. Dylla (2015). Learning relational event models from video. *Journal of Artificial Intelligence Research 53*, 41–90.

Dumas, B., D. Lalanne, and S. Oviatt (2009). Multimodal interfaces: A survey of principles, models and frameworks. *Human machine interaction*, 3–26.

Eisenstein, J., R. Barzilay, and R. Davis (2008a). Discourse topic and gestural form. In *AAAI*, pp. 836–841.

Eisenstein, J., R. Barzilay, and R. Davis (2008b). Gesture salience as a hidden variable for coreference resolution and keyframe extraction. *Journal of Artificial Intelligence Research 31*, 353–398.

Fussell, S. R., R. E. Kraut, and J. Siegel (2000). Coordination of Communication: Effects of Shared Visual Context on Collaborative Work. In *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, CSCW '00, New York, NY, USA, pp. 21–30. ACM.

Fussell, S. R., L. D. Setlock, J. Yang, J. Ou, E. Mauer, and A. D. I. Kramer (2004, September). Gestures over Video Streams to Support Remote Collaboration on Physical Tasks. *Hum.-Comput. Interact. 19*(3), 273–309.

Gergle, D., R. E. Kraut, and S. R. Fussell (2004). Action As Language in a Shared Visual Space. In *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work*, CSCW '04, New York, NY, USA, pp. 487–496. ACM.

Goldstone, W. (2009). *Unity Game Development Essentials*. Packt Publishing Ltd.

He, K., X. Zhang, S. Ren, and J. Sun (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778.

Jaimes, A. and N. Sebe (2007). Multimodal human–computer interaction: A survey. *Computer vision and image understanding 108*(1), 116–134.

Kennington, C., S. Kousidis, and D. Schlangen (2013). Interpreting situated dialogue utterances: an update model that uses speech, gaze, and gesture information. *Proceedings of SIGdial 2013*.

Kraut, R. E., S. R. Fussell, and J. Siegel (2003, June). Visual Information As a Conversational Resource in Collaborative Physical Tasks. *Hum.-Comput. Interact. 18*(1), 13–49.

Krishnaswamy, N. and J. Pustejovsky (2016a). Multimodal semantic simulations of linguistically underspecified motion events. In *Spatial Cognition X: International Conference on Spatial Cognition*. Springer.

Krishnaswamy, N. and J. Pustejovsky (2016b). VoxSim: A visual platform for modeling motion language. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. ACL.

Lascarides, A. and M. Stone (2006). Formal semantics for iconic gesture. In *Proceedings of the 10th Workshop on the Semantics and Pragmatics of Dialogue (BRANDIAL)*, pp. 64–71.

Lascarides, A. and M. Stone (2009a). Discourse coherence and gesture interpretation. *Gesture 9*(2), 147–180.

Lascarides, A. and M. Stone (2009b). A formal semantic analysis of gesture. *Journal of Semantics*, ffp004.

Madeo, R. C. B., S. M. Peres, and C. A. de Moraes Lima (2016). Gesture phase segmentation using support vector machines. *Expert Systems with Applications 56*, 100–115.

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

McDonald, D. and J. Pustejovsky (2014). On the representation of inferences and their lexicalization. In *Advances in Cognitive Systems*, Volume 3.

Mehlmann, G., M. Häring, K. Janowski, T. Baur, P. Gebhard, and E. André (2014). Exploring a Model of Gaze for Grounding in Multimodal HRI. In *Proceedings of the 16th International Conference on Multimodal Interaction*, ICMI '14, New York, NY, USA, pp. 247–254. ACM.

Paul, R., J. Arkin, N. Roy, and T. M. Howard (2016). Efficient grounding of abstract spatial concepts for natural language interaction with robot manipulators. In *Robotics: Science and Systems*.

Pustejovsky, J. (1995). *The Generative Lexicon*. MIT Press, Cambridge, MA.

Pustejovsky, J. (2013). Dynamic event structure and habitat theory. In *Proceedings of the 6th International Conference on Generative Approaches to the Lexicon (GL2013)*, pp. 1–10. ACL.

Pustejovsky, J. and N. Krishnaswamy (2016, May). VoxML: A visualization modeling language. In N. C. C. Chair), K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis (Eds.), *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France. European Language Resources Association (ELRA).

Pustejovsky, J. and J. Moszkowicz (2011). The qualitative spatial dynamics of motion. *The Journal of Spatial Cognition and Computation*.

Quek, F., D. McNeill, R. Bryll, S. Duncan, X.-F. Ma, C. Kirbas, K. E. McCullough, and R. Ansari (2002). Multimodal human discourse: gesture and speech. *ACM Transactions on Computer-Human Interaction (TOCHI) 9*(3), 171–193.

Skantze, G., A. Hjalmarsson, and C. Oertel (2014, November). Turn-taking, feedback and joint attention in situated human-robot interaction. *Speech Communication 65*, 50–66.

Turk, M. (2014). Multimodal interaction: A review. *Pattern Recognition Letters 36*, 189–195.

Wang, I., P. Narayana, D. Patil, G. Mulay, R. Bangar, B. Draper, R. Beveridge, and J. Ruiz (2017a). EGGNOG: A continuous, multi-modal data set of naturally occurring gestures with ground truth labels. In *To appear in the Proceedings of the 12th IEEE International Conference on Automatic Face & Gesture Recognition*.

Wang, I., P. Narayana, D. Patil, G. Mulay, R. Bangar, B. Draper, R. Beveridge, and J. Ruiz (2017b). Exploring the use of gesture in collaborative tasks. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '17, New York, NY, USA, pp. 2990–2997. ACM.

Wobbrock, J. O., M. R. Morris, and A. D. Wilson (2009). User-defined Gestures for Surface Computing. CHI '09, New York, NY, USA, pp. 1083–1092. ACM.

Zhang, Z. (2012). Microsoft kinect sensor and its effect. *IEEE MulitMedia 19*, 4–10.