A Real-time Robotic Model of Human Reference Resolution using Visual Constraints

Matthias Scheutz(*) Kathleen Eberhard(**)
Virgil Andronache(*)
(*) Department of Computer Science and Engineering
(**) Department of Psychology
University of Notre Dame, Notre Dame, IN 46556, USA
Phone: +1 574 631-0353
Fax: +1 574 631-9260
Email: {mscheutz,keberhar,vandrona}@nd.edu

Abstract

Evidence from recent psycholinguistic experiments suggests that humans resolve reference incrementally in the presence of constraining visual context. In this paper, we present and evaluate a computational model of human reference resolution that directly builds a semantic interpretation of an utterance without the need for a separate syntactic analysis phase, which typically involves the construction of parse trees. The model is implemented on a robot using real audio and video inputs, thus operates in real-time, and is distributed over several computers, which run in parallel. Results from experiments with the model confirm the viability of the algorithm to process semantic interpretations, in particular, reference incrementally, as demonstrated to be employed by humans.

A Real-time Robotic Model of Human Reference Resolution using Visual Constraints

1 Introduction

The classic view of human language processing started with Chomsky (1957, 1965, 1986) and was subsequently adopted by the artificial intelligence community, e.g., through Winograd (1972, 1973) and others. It assumes that human language processing proceeds in stages, which include at least a syntactic and a semantic stage. In particular, at the syntactic stage, parse-tree representations of the sentence are produced, which provide the syntactic categories of the expected words in order. If a recognized word violates the expected syntactic category, the current parse tree will be discarded and a new alternative parse tree will be built. Similarly, a new parse is constructed if the semantic analysis reveals or suggests that the current parse tree must be incorrect.

Although this approach seems plausible, recent evidence from psycholinguistic studies of human sentence processing suggests that parse trees may contribute minimally or not at all to comprehension in communicative situations in which the referential context is visually co-present with both the listener and the speaker, and, therefore, highly accessible; i.e., it does not need to be maintained in working memory (e.g., Chambers, Tanenhaus, Eberhard, Filip, & Carlson, 2002; Eberhard, Spivey-Knowlton, Sedivy, & Tanenhaus, 1995; Kamide, Altmann, & Haywood, 2003; Tanenhaus, Spivey-Knowlton, Eberhard, & Sedivy, 1995; Sedivy, 2002). As summarized further in the next section, under these conditions, listeners rapidly and incrementally interpret spoken utterances with respect to the visual context establishing reference as soon as the utterance provides sufficient information for distinguishing the intended referent from alternatives, even when this information occurs before the end of the syntactic constituent. The evidence of rapid incremental integration of linguistic and contextual information is consistent with a constraint-satisfaction view of human sentence processing (e.g., MacDonald, Pearlmutter, & Seidenberg, 1994; Tanenhaus & Trueswell, 1995). However, the findings have yet to be explicitly modeled within this theoretical framework.

In this paper, we present a *parallel, distributed, real-time, robotic processing model of human reference resolution* that uses context information to disambiguate referential expressions to determine the semantic content of natural language sentences without the use of syntax trees. Specifically, the model operates in a real blocks' world-like environment and can resolve the reference of referring expressions when presented with spoken sentences like "Put the red block on the green block on the blue block" in real-time. More importantly, the model shows the same performance in the cases where humans can quickly resolve reference, while also exhibiting the same comprehension difficulties observed in humans when the referring expressions are ambiguous due to underspecification or overspecification with respect to a given context.

The paper is organized as follows: first we provide some background on human reference resolution from a psychological perspective and summarize the findings which suggest that humans resolve reference incrementally utilizing as much context information as possible. We then present the functional architecture of our computational model, including its core and marginal assumptions. After a brief description of the experimental setup, we show the results of several experimental runs with the model, which follow the same pattern as those of the human subjects.

We end with a discussion of the implications of the model for natural language processing and with a prediction that we derive from the model for further psychological experiments.

2 Human Sentence Processing

Studies of human sentence processing show that comprehension occurs incrementally on essentially a word-by-word basis (e.g., Steedman, 1989; Tanenhaus & Trueswell, 1995). One reflection of this incremental interpretation is the occasional occurrence of misinterpretations–or garden paths, as illustrated by the sentence, "Joe loaded the boxes on the cart into the van", in which the first prepositional phrase (PP) "on the cart" is often initially misinterpreted as specifying the goal of the loading event rather than specifying the location of the theme (the boxes) or object of the loading event. Studies in which readers' eye movements are recorded as they read sentences show that the misinterpretation is often reflected in long fixation durations on the disambiguating portion of the sentence (e.g., on the second PP "into the van", in the example above) as well as occasional regressive eye-movements, reflecting a re-reading of the misinterpreted portion of the sentence.

Factors that contribute to or attenuate the occurrence of garden paths are a primary emphasis in much of psycholinguistic research on human sentence comprehension. Specifically, theories about the underlying processes can be roughly classified as either *two-stage* or *one-stage* theories. Two-stage theories (e.g., Clifton, Speer, & Abney, 1991; Kimball, 1973; Ferreira & Clifton, 1986; Frazier & Fodor, 1978; Frazier, 1995 assume that the initial or first stage of comprehension occurs strictly on the basis of the grammatical categories of words, phrase-structure rules for combining categories into phrases, and a set of parsing principles designating which rule is applied in the event that more than one is applicable. The second stage, in turn, assigns semantic roles such as agent, theme, goal, etc. to the phrasal constituents based on their attachment in the parse tree. Thus, according to this view, encountering "on" in the example sentence above triggers the construction of a PP constituent, which, according to the phrase-structure rules of English, can be attached to two possible positions in the parse tree. It can be attached to the noun phrase (NP) node, corresponding to the theme "the boxes", in which case it would be assigned the semantic role of locative (specifying the location of the theme) during the second stage. Alternatively, the PP can be attached to the higher verb phrase (VP) node, in which case it would be assigned the semantic role of goal at the second stage. A *Minimal Attachment* principle, which stipulates building the simplest parse tree, resolves the ambiguity in favor of attaching the PP to the higher VP node, where it is initially assigned, and, hence interpreted as the goal. An error in this initial interpretation is signaled when the second preposition "into" is encountered, which triggers the construction of another PP that must be attached to the higher VP node. This error triggers a re-analysis process in which the initial attachment of the first PP is revised so that it is attached to the NP node, where it is appropriately assigned the locative role.

In contrast, the one-stage theory assumes that the initial incremental interpretation of a sentence utilizes all sources of information including, syntactic, semantic, pragmatic, etc. Garden-paths occur because the various sources of information often differ in their relative availability, particularly as a sentence unfolds. Specifically, bottom-up or *local* sources of information, such as the syntactic category of words and their lexical-semantic requirements (e.g., the semantic arguments associated with verbs) are generally more accessible than top-down or *global* sources, such as the pragmatic or discourse context in which a sentence is embedded. Thus, most of the empirical tests of the opposing theoretical views have focused on whether top-down contextual constraints can prevent the initial occurrence of a garden path (e.g., Frazier, 1995; MacDonald et al., 1994).

For example, a study by Ferreira and Clifton (1986) recorded readers' eye movements as they read sentences such as, "Joe loaded the boxes on the cart into the van", which were preceded by a sentence that established two possible referents in the discourse context for the definite NP corresponding to the theme in the critical sentence (i.e., two sets of boxes). According to the one-stage view, this ambiguous reference should prevent a garden path by requiring the initial interpretation of the first PP as a locative (i.e., specifying which set of boxes was the referent of the definite NP) rather than as the goal.

The results, however, supported the two-stage view by showing evidence of garden paths despite the presence of a context. Nevertheless, in reading studies such as Ferreira and Clifton's, the discourse context must be maintained in working memory, and, therefore, it may not be readily accessible to bias the initial incremental interpretation of a sentence. A stronger test of whether a discourse context can be used to avoid a garden path comes from more recent studies that have examinined the incremental interpretation of spoken sentences that refer to visually co-present referents.

2.1 Eye-Movements as an Online Measure of Spoken Language Comprehension

Tanenhaus, Spivey-Knowlton, Eberhard, & Sedivy (Tanenhaus et al., 1995; Eberhard et al., 1995) developed an eye-movement recording method that provides a sensitive and continuous measure of listeners' incremental interpretation of spoken sentences. Specifically, their method involved recording listeners' eye-movements as they followed spoken instructions for moving common objects on a display table (e.g., "Put the napkin in the bowl."). Eye-movements were recorded via a light-weight eye-tracking camera that was attached to an adjustable headband worn by the listener. The headband also contained a small video camera that recorded the visual scene from the listener's perspective. The scene image was displayed on a TV monitor along with a record of the listener's eye fixations, which were superimposed as cross hairs. Both the image on the TV monitor and the experimenter's spoken instructions were recorded by a Hi8 VCR that permitted frame-by-frame playback of the synchronized audio and visual channels. Analyses of the video tapes involved logging the location and duration of all fixations that occurred relative to onsets and offsets of the words in the critical spoken instructions.

The methodology took advantage of the fact that when given an instruction such as "Put the napkin in the bowl", a listener will naturally and automatically fixate the to-be-moved object (napkin) before reaching for it because the motor system that programs the reach for the object requires information from the visual system about the object's location. Thus, a listener's fixation on an object that occurs prior to his or her reach for it is a behavioral index of when he or she has identified the referent that serves as the "theme" of the putting action. Initial studies using this technique demonstrated that the listeners' fixations are remarkably time-locked to the spoken referring expressions, with the fixation that precedes the reach to an object occurring as soon as sufficient information has been provided for identifying it as the intended referent, even when that information occurs before the end of the referring expression.

2.2 Evidence for the Online Incremental Use of Context to Resolve Reference

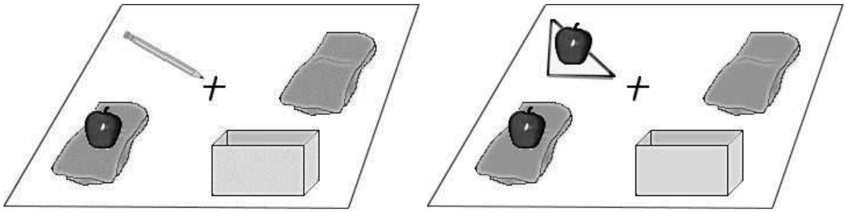


Figure 1: Examples of a one-referent and two-referent display context, respectively.

Tanenhaus et al. (Eberhard et al., 1995; Spivey, Tanenhaus, Eberhard, & Sedivy, 2002) used the eye-movement recording methodology to test the contrasting predictions of the two-stage versus one-stage (incremental) views. In particular, on each trial listeners heard a series of spoken instructions that directed them to put one or more *theme* objects on the left side of a display table in or on one or more *goal* objects on the right side of the table. The trials varied the kinds of theme and goal objects as well as the kinds of actions that were requested. The critical trials represented a complete crossing of two linguistic factors with two display context factors. As shown by the example sentences in (1a) and (1b) below, the two linguistic factors were whether the critical sentences were syntactically ambiguous (1a) or syntactically unambiguous (1b).

(1a) *Syntactically ambiguous:* Put the apple on the towel in the box.

(1b) *Syntactically unambiguous:* Put the apple that is on the towel in the box.

Both types of critical instructions were presented in a *one-referent* display context condition (shown on the left in Figure 1), and a *two-referent* display context (shown on the right in Figure 1). Specifically, the one-referent context contained only one possible theme referent (i.e., one apple), whereas the two-referent context contained two possible theme referents (i.e., two apples, one on a towel and another on a napkin). Both display contexts supported an interpretation of the first PP in the ambiguous instruction (i.e., “on the towel”) as either a locative specifying the theme (i.e., the location of the apple that was to be moved) or a goal of the “put” action. The correct interpretation corresponded to the locative assignment, which was the assignment required by the syntactically unambiguous instruction.

Of interest was whether the listeners’ initial interpretation of the ambiguous instruction would differ from their initial interpretation of the unambiguous instruction, as reflected in a difference in the pattern of their eye fixations and reaches that occurred when the two instructions were presented in the one-referent versus two-referent conditions.

2.3 Results

Figure 2 shows the typical sequence of fixations on objects in the display when the ambiguous and unambiguous instructions were given in the one-refernt context (left side) and in the two-referent context (right side).

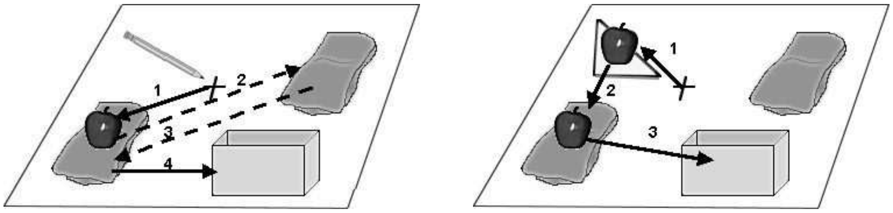


Figure 2: Typical sequence of eye movements to the objects in the one-referent (left) and two-referent (right) display contexts for the ambiguous and unambiguous instructions. The solid arrows indicate eye movements that occurred during both the ambiguous and unambiguous instructions. The dashed arrows indicate the additional eye movements that occurred during the ambiguous instruction in the two-referent context.

The numbers in the instructions below show the occurrence of the fixations that are indicated by the corresponding numbered arrows in the one-referent context of Figure 2.

(1a) *Syntactically ambiguous:* Put the apple on the (1) towel in (2) the box. (3) (4)

(1b) *Syntactically unambiguous:* Put the apple that is (1) on the towel in the box. (4)

Consistent with both the one- and two-stage views' predictions, the timing and number of fixations that occurred during the instructions differed for the ambiguous and unambiguous instructions. In particular, the listeners' fixations during the ambiguous instruction indicated a significant tendency to initially *misinterpret* the first PP as specifying the goal of the putting action as reflected by a significant number of fixations on the towel on the right side of the display (arrow #2 in Figure 2) shortly after the PP "on the towel". This initial interpretation had to be revised after the second PP, "in the box", which was reflected in the occurrence of two more fixations: one on the apple (arrow #3) and then one on the box (arrow #4), which was the correct goal. In contrast, when the unambiguous instruction was given in the one-referent context only two fixations occurred. Like the ambiguous instruction, the first fixation (arrow #1) was to the single apple in the display and occurred shortly after the NP "the apple". The second and last fixation (arrow #4) was on the box and occurred shortly after the second PP "in the box".

The two-stage view attributes the misinterpretation of the first PP in the ambiguous instruction to the use of a *Minimal Attachment Principle* to resolve the syntactic ambiguity during the first stage of parsing. The error in this resolution, which is signalled by the uncertainty of the semantic role assignment for the second PP, triggers a reanalysis process that revises the initial assignment of goal to the first PP. The one-stage view, in contrast, attributes the misinterpretation of the first PP to the use of pragmatic constraints about the kind of information that a listener expects to receive from a "cooperative" speaker. Specifically, according to *Gricean Maxims* (Grice, 1975), speakers are tacitly expected to provide only the necessary and sufficient information for identifying their intended referents. Because the definite NP "the apple" provided sufficient information for identifying the theme, the first PP was initially interpreted as specifying the goal of the putting action, as opposed to providing (superfluous) information for identifying the theme.

The numbers in the instructions below summarize the results when the instructions were given in the two-referent condition (shown on the right in Figure 2). Like the sentences above, the numbers show the occurrence of the fixations that are indicated by the corresponding numbered arrows for this condition Figure 2.

(1a) *Syntactically ambiguous:* Put the apple on the (1) towel (2) in the box. (3)

(1b) *Syntactically unambiguous:* Put the apple that is (1) on the towel (2) in the box. (3)

The sequences of fixations in this condition supported the one-stage view's predictions. Specifically, the number and timing of the fixations were the same for the ambiguous and unambiguous instructions, providing evidence that both instructions were initially interpreted in the same manner. Crucially, for both instructions, there was a negligible number of looks to the towel on the right side of the display after the first PP "on the towel". Instead, for both instructions, if the first fixation after the NP "the apple" was on the incorrect apple (the apple on the napkin), then there was a second fixation on the correct apple shortly after the PP "on the towel". The final fixation (arrow #3) in both instructions occurred on the box shortly after the second PP "in the box".

Thus, contrary to the previous results from reading comprehension studies, Tanenhaus et al.'s results showed that when the referential context is visually co-present and does not need to be maintained in working memory, it is immediately used by the incremental interpretive processes to resolve reference.

3 A Parallel Distributed Processing Model of Human Reference Resolution

In this section we develop the details of our model of incremental human reference resolution and give a functional description of the model. Implementation details (such as how functional components get mapped onto computer hardware) will be addressed in the next section. We believe that it is important to separate the functional from the implementation level, for it is possible to implement any functional specification in different ways using different underlying systems (e.g., symbolic algorithms or neural nets) and looking at the underlying system might obscure important features of a model that do not depend on implementation details.

We start with a description of the domain, in which we will evaluate our model, and give examples of the kinds of sentences the model can encounter and needs to be able to cope with.

3.1 Reference Resolution in a Blocks World Domain

The domain under consideration is a simple *Blocks World Domain*, in which blocks of different color can be stacked (e.g., see the two examples in Figure 3). Blocks can exhibit one of the following relationships: they can be *on*, *under*, or *next-to* other blocks, or they can be isolated without any immediately adjacent block. In this domain, it is easily possible to arrange situations, where referential expressions are ambiguous or overdetermined. For example, the phrase "the red block" fails to pick out a referent if there are two or more red blocks in a scene (as on the right in Figure 3). However, if "the red block" is followed by "on the orange block", and if there is only

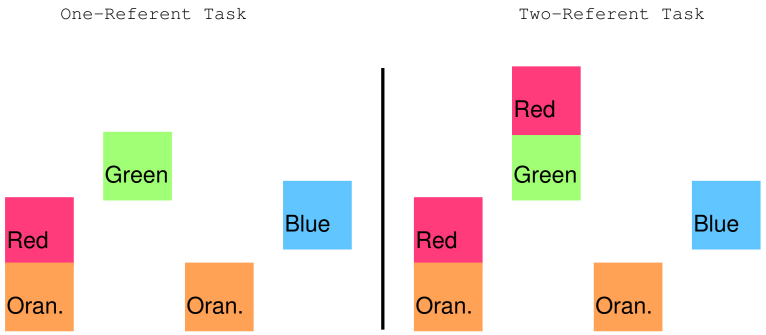


Figure 3: The overall setup of the experiment for the resolution of referents in instructions like “Put the red block on the orange block on the blue block.”

one red block on an orange block, the unique reference can be established. Similarly, a phrase like “the red block on the orange block” is an overdetermination if there is only one red block (on a orange block) in the scene (as on the left in Figure 3).

In our model we will consider sentences of the form “Put ... on ... on ...”; for example, “Put the red block on the orange block on the blue block”. Syntactically, the example sentence has two parses (see Figure 4).

Logically, these two parses correspond to two different semantic interpretations, which can be expressed as follows (we abbreviate the complex predicate expressions like “$\text{Red}(x) \wedge \text{Block}(x)$” by “$\text{RedBlock}(x)$”):

(I1) $\text{PUT}((\iota x)\text{ON}(\text{RedBlock}(x),\text{OrangeBlock}(x)), (\iota x)\text{BlueBlock}(x))$

(I2) $\text{PUT}((\iota x)\text{RedBlock}(x), (\iota x)\text{ON}(\text{OrangeBlock}(x),\text{BlueBlock}(x)))$

If presented with either environment, a classical parser will typically start with the bottom parse tree in Figure 4, and hence will be forced in both cases to reject its first choice. Consider, for example, the one-referent condition in Figure 3. Starting with the bottom parse tree, the attempt to resolve the reference of “the orange block on the blue block” fails, because the semantic interpretation of that phrase, $(\iota x)\text{ON}(\text{OrangeBlock}(x),\text{BlueBlock}(x))$ is false in the scenario. Consequently, in a typical computational approach where new parse trees are generated until they can be assigned a possible semantic interpretation, this parse tree will be discarded and another parse tree will be considered (if it exists). In the above case, there is another parse tree (top in Figure 4), which requires that $(\iota x)\text{ON}(\text{RedBlock}(x),\text{OrangeBlock}(x))$ be true, which it is in the given scenario. Thus, the second parse succeeds.

The method of determining the meaning of sentences as described is essentially based on the idea that parse trees are generated before an interpretation of the involved terms is attempted. Consequently, when it is not possible to find a consistent interpretation for a generated parse tree in a given scenario, the syntactic analysis will have to be revisited. In the best case, a *backtracking process* will allow the system to isolate the *nearest choice point* in the generation process of the current parse tree, where a decision was made regarding a particular way of parsing it (e.g., because there were two syntax rules with the same left-hand side, but different right-hand sides, which can

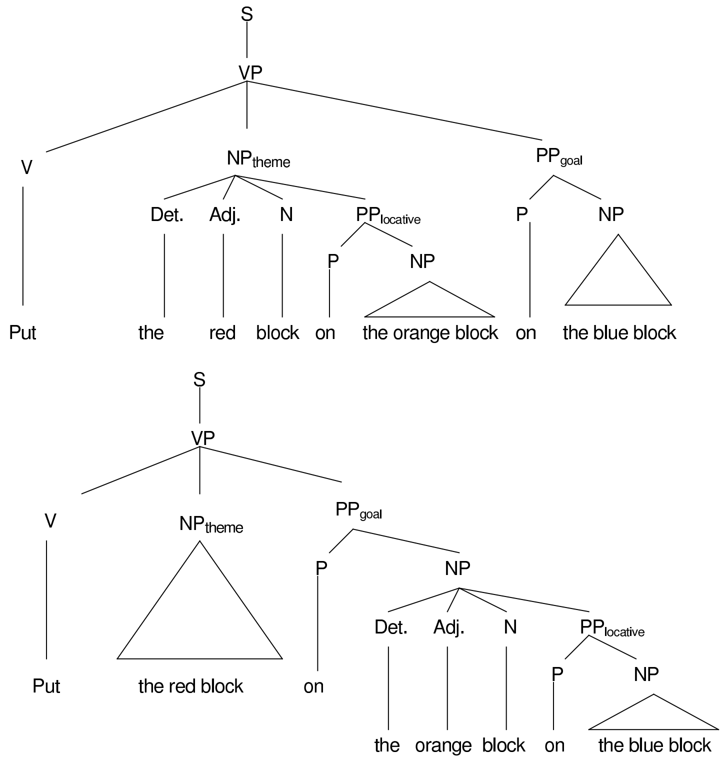


Figure 4: The two parses of the sentence "Put the red block on the orange block on the blue block".

occur in a non-deterministic context-free grammar). At the choice point, parsing can then resume applying a different grammar rule. While the backtracking process thus may allow the system to preserve partially built semantic structures and only rebuild those that caused the failure of completing the semantic interpretation of the parse tree, in the worst case the root of the parse tree will be reached and a complete new parse will have to be built. Consequently, any (partial) semantic interpretation might have to be discarded.

As argued in Section 2, algorithms that separate syntactic and semantic analysis by first building a parse tree and then attempting to assign a consistent interpretation to its lexical constituent parts do not seem to capture the processes used by humans to resolve reference, at least not in the context of visual constraints. Rather than always using parse trees, humans seem to be able to resolve reference sometimes incrementally without having to construct explicit parse trees and use backtracking to produce alternative parse trees only if the incremental parse fails.

We would, therefore, like to propose the following alternative picture. Consider again the phrase "the red block on the orange block". Upon hearing "red" in the above sentence and scenario, we hypothesize that humans construct a set of representations of possible referents **PR**(RED), which at this point contains representations of the two red blocks. Since the set is not a singleton, the subsequent words will be taken to specify constraints on **PR** and thus to belong to the same referential phrase. Hence, upon hearing "on" a constraint will be instantiated that limits the referents in **PR**(RED) based on the set **PR**(ORANGE), which will be formed upon hearing "orange". Since the newly formed set **PR**(ORANGE) is a singleton, the referent, which is the argument for "put" has been established to be the single element in **PR**(ORANGE). Hence, subsequent words are not taken to be part of the definite description anymore.

The above example demonstrates that the referents of definite descriptions can be determined incrementally using visual constraints without constructing parse trees. According to the incremental approach, the syntactic ambiguity does not affect the semantic process of establishing reference because the semantic process does *not* rely on the syntactic structure of the sentence. Rather, the semantic process simply interprets each word as it is encountered with respect to the context in which it occurs and uses that interpretation to further reduce the set of possible referents to a single unique one. When a unique referent is identified it is assigned the appropriate thematic role (e.g., theme) and the semantic process moves on to identify the next referent that needs to be assigned a thematic role, where the assignment of thematic roles is, in large part, determined by the semantics of the verb (e.g., "Put" is a 2-place predicate that requires a theme argument and a goal argument). Essentially, this approach is based on the following

> *Minimal Information Processing Principle:* A human speaker will typically produce sentences in such a way as to provide the minimum information necessary to identify his or her intended referent. To do this, the speaker must take into account the domain of reference or set of entities that are likeky to be considered as possible referents by the listener. This domain of reference is part of the mutual or shared knowledge, also called "common ground", that is used by speakers and listeners to communicate messages efficiently, e.g., (Clark, 1996; Karttunen & Peters, 1975; Stalnaker, 1978).

By the same token, listeners expect only the minimum information necessary given the domain of reference to identify the intended referent.

From a pragmatic perspective, the *Minimal Information Processing Principle* can be thought of as a "communication protocol" that tacitly underwrites human communicative interactions (see Grice's maxims, in particular, the "Maxims on Quantity", Grice, 1975 as well as the *Principle of Relevance* by Sperber & Wilson, 1995). Hence, sentences that violate this principle, either by overspecifying reference or underspecifying reference are likely to increase comprehension difficulty for the listener. Yet, they might not cause problems for a classic approach to parsing and language understanding as outline above. Specifically, humans will have comprehension difficulties in some cases of referential *overdetermination*, whereas the classic approach will (eventually) find the right parse and thus the appropriate interpretation.

To see this, consider the situation depicted on the left in Figure 3, where the referent of "the red block" is uniquely specified. Taking again the above sentence "Put the red block on the orange block on the blue block", a classic parser would eventually produce the only parse consistent with the context (i.e., on the top in Figure 4), and thus interpretation (I1). Hence, there is no ambiguity in the semantic interpretation of the sentence. Yet, for human subjects this sentence together with the given setup of blocks causes problems, because it violates the *Minimal Information Principle*, which underwrites, as we would like to suggest, the method by which humans determine reference (cf. again to Section 2). For according to the above-described, hypothesized algorithm for human reference resolution the reference to a unique object is established upon hearing "red", at which point the term "on" is not taken to belong to the same phrase as "the red block" anymore, but rather is assumed to be associated with "put" (as an indicator that the goal of the *put* action will now be specified). Consequently, the second argument of "put", which is the goal argument, is taken to be the referent of "orange block". Although this is a set with two possible referents, the precondition of "put"–that no block can be on top of a target of the *put action*–eliminates the orange block underneath the red block as a possible referent, and thus reduces the set of possible referents for "orange" to a singleton (namely, the orange block on the right).[1] Hence, at this point, the minimal information is present that allows human subjects to carry out the put action. However, the sentence continues with the phase "on the blue block", of which the human subjects attempt to make sense. This requires them them to revise their semantic interpretation so that it either corresponds to the intended interpretation in (I1), or that is corresponds to the unintended interpretation in (I2). The latter requires them to further alter the context by placing the red block on the organge block together on the blue block (several subjects in the experiments in Tanenhaus et al. (1995) resortet to this latter strategy).

3.2 The High-Level Functional Architecture

We are now ready to sketch the high-level functional architecture of a model, which is intended to capture the above-described idea of incrementally restricting the sets of possible referents of an expression until the set only contains one member, the *referent of the expression*. The model is a hybrid symbolic-connectionist model based on the idea of parallel distributed processing, where multiple processing units are concurrently active, exchange information via communication links, and constrain each others' activations.[2] At a high level of description the model architecture can be broken down into twelve functional components, some of which are connected to visual and auditory sensors and to the camera motor effector (see Figure 5).

Each rectangle indicates a component type, whose function is indicated by its label. While some types only have one instance in the running virtual machine (e.g., speech recognition), others can have multiple ones (e.g., blob tracking). Auditory and visual sensory information is transformed by transducers (not shown) and enters into the architecture in the perceptual processing components (i.e., the speech recognition and color/shape recognition modules). Lexical representations are then analyzed as individual words (as opposed to syllables or morpho-syntactic units, say). Each recognized word is looked up in a dictionary to obtain its semantic form (if there is one), and a corresponding processing unit is instantiated for further processing. For example, if the word "put" is encountered, then processing unit for "put" is instantiated, which subsequently processes its argument types (i.e., the object and location of the "put" action). If a referent is needed at any given time (e.g., as part of the processing of an action term unit), then a visual search process is initiated to find the referent in the visual scene (e.g., the referent of "the blue block").[3] Visual search involves looking through a list of already tracked objects as well as trying to find new objects using color blobs and shape detection. It may also involve camera movements if the objects are not entirely within visual range. For each simple object that meets the requirement, a tracker is

[1] This precondition could viewed as being derived from the semantics of "put", which requires a change of location to take place, which would not occur with the orange block below the red block being construed as the goal.

[2] While neural networks have come to be viewed as prototypical simple instances of such systems (McClelland & Rumelhart, 1988), we believe that the notion of "parallel distributed processing" should not be restricted to simple processing units with numeric data, but should be applied more generally to any kind of processing architecture, in which multiple, possibly heterogeneous units process information in parallel in a distributed fashion.

[3] To simplify perceptual processing, we do not use 3D blocks, but use a possible 2D projection, i.e., a square.

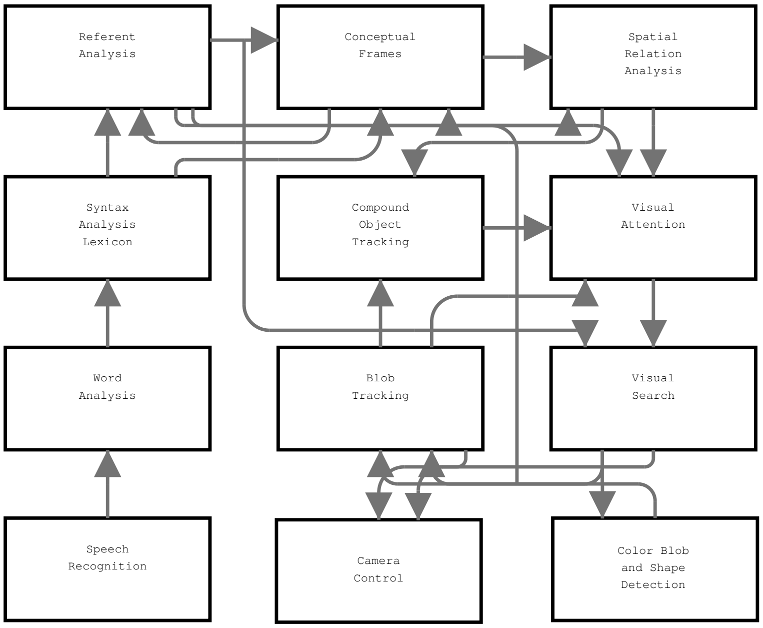


Figure 5: A high-level view of the components employed in the architecture used in the reference resolution task.

instantiated that encapsulates its basic properties (e.g., color and shape) and automatically tracks the object (e.g., if they should move or the camera should move).

Processing complex objects is more involved. For example, finding the referent of "the blue block on the red block" will involve finding blue squares first, then finding all red squares and performing a spatial relation analysis on each of the red squares to see if the "on" relation holds (i.e., if a red square is below the blue square). As part of this process, compound trackers will be instantiated that form and maintain representations of spatial relationships. Compound trackers track their individual parts by virtue of other compound or simple trackers. For example, a compound tracker corresponding to "the red block on the green block to the left of the red sphere" might consist of a compound tracker for "to the left of", which takes as its first argument another compound tracker (for "the red block on the green block") and as its second argument simple tracker (for "the red sphere"). At any time, visual attention will be directed at uniquely identified objects (as denoted by definite descriptions) or will focus on one of multiple possible referents (typically, the one closest in visual space to the previous focus).

The model, as described above, makes several assumptions about information processing in humans. While some of these assumptions are based on psychological evidence (e.g., assumptions about eye movements and focus of attention), most of them are of hypothetical nature and can only be indirectly verified by comparing the model's behavior to that of humans. Specifically, we distinguish two classes of assumptions: *core assumptions* that are essential to how the model works, and *marginal assumptions* that can be modified without invalidating the model.

3.3 Core Assumptions

We start with the eight core assumptions:

1. *Default incremental sentence processing:* Sentences are by default processed incrementally based on the input stream of lexical items, i.e., for each recognized lexical item a direct semantic interpretation is attempted without first analyzing the overall syntactic structure of the sentence.

2. *Lexical input stream:* The lexical input stream will be continuously monitored by a lexicon processing unit that maps lexical items to semantic types and automatically instantiates the corresponding processing units unless other units take control of the input stream.

3. *Control flow:* Each processing unit that obtains control of the lexical input stream processes lexical items specific to its function from the stream until a final processing state is reached or temporarily passes control to other processing units (e.g., an action term processing unit passes control to a definite description processing unit to get its arguments).

4. *Default incremental reference resolution:* The reference relation between terms and an (intended) objects is established incrementally by a computational unit checking for definite descriptions, which will get invoked by processing units that require referents, keep control until reference is established or a parsing error is encountered, and return control to the invoking unit.[4]

5. *Set of possible referents:* The extension of a simple or complex term d at any time t is a set of objects or "possible referents" **PR** that the term could denote given the information available to the system up to time t (e.g., the set of all red blocks for the term "block").

6. *Establishment of reference:* The reference of a simple or complex term d (to an object) is established when the term's set of referents **PR** has only one member.

7. *"That"-clauses:* An action processing will return control to the unit processing definite descriptions if a "that"-clause is encountered after a referring expression was successfully processed and a referent was determined assuming that the "that"-clause will further specify the previously determined referent (e.g., it could "overspecify" a referent as in "the red block that is on the orange block" on the right in Figure 3).

8. *Action term processing:* An action term d will instantiate directly an action processing unit (whose processing details will depend on the semantics of d), which subsequently will instantiate definite description processing units to determine its argument token(s) (e.g., the object(s) to be manipulated or the location(s) at which an action should occur).

[4] In the case of a parsing error, the involved unit will signal an error to its invoking instance. Local error recovery will be attempted, and if it does not succeed, the error will be propagated further up the instantiation hierarchy until it reaches supervisory control–a detailed account of error processing and recovery from parsing failures will have to await another occasion.

The first three assumptions are about the overall processing architecture and how distributed processing occurs in the system, the next five assumptions concern the details of processing referential expressions, and the last assumption deals with the direct semantic processing of verbs denoting actions. Changes to any of these assumptions might (partially) invalidate the model, and may or may not yield the predicted results.

3.4 Marginal Assumptions

The seven marginal assumptions are:

1. *Incremental lexical processing:* Incremental processing of referents occurs at the word level (if words denoting possible referents have no initial syllables is common).

2. *Focus of attention:* Whenever a new set of referents R (with $|R| > 1$) is transferred to the attention module, the eye will fixate referent r such that $||r - c|| \leq ||x - c||$ for all $x \neq r \in R$ (where c denotes the current fixation of the eye in the visual scene and $||.||$ denotes the Euclidean distance in the image plane).

3. *Overdetermination:* The focus of attention is lost if reference is overspecified.

4. *Ambiguous reference:* If reference is ambiguous, a referent is chosen at random (typically, the one "closest" to the current focus of attention).

5. *Common workspace:* All processing units and representations created as part of the lexical processing are instantiated in a common workspace that can be accessed by all units that require access (as fixed by the architecture description).

6. *Visual processing:* For each (possible) referent of a term from the lexical input stream a simple tracking unit is instantiated that attempts to track the object in the visual image and a complex tracking unit is instantiated to track simple trackers whose tracked objects are arranged in particular spatial relations (e.g., the complex "on" tracker can track an object as long as it is *on* another object).

The difference between the core and marginal assumptions is that the marginal assumptions can be changed in different ways without affecting the validity of the model, although changes may affect the way the model is tested. For example, if assumptions about the focus of attention are changed, then it may not be possible to determine based on focus of attention alone, whether the model has successfully processed a sentence.

4 Model Evaluation

The experimental evaluation of the model is intended to test the model's ability to replicate the findings in the human case: that overdetermination causes comprehension difficulties which are assumed to be due to the particular way in which humans resolve reference incrementally in the presence of constraining visual contexts. Specifically, the model should be able to "comprehend" the same set of sentences that humans comprehend and show comprehension difficulty where humans have problems. “Comprehension” as well as “comprehension difficulty” are measured using the model’s visual attention mechanism: analogous to the human case, where eye movements are observed in line with the processed lexical items–the eyes move from possible referent to possible referent based on the state of processing–the focus of attention module determines what item is processed at any given time and reflects the (partial) internal representation of the sentence at any given point.

Beyond replicating the results from human findings, the model can also be evaluated with respect to three practical features that distinguish it from many other cognitive models:

- *Embodiment:* whether it can process information reliably under different real-world conditions (e.g., different lighting or background noise conditions, different speaker, different angle to display, etc.)
- *Real-time:* whether it can perform the parsing and building of semantic representations for different sentence types in real-time
- *Distributed parallel processing:* whether it can run distributed on multiple computers in real parallelism (i.e., on different CPUs rather than in “simulated parallelism” on one CPU)

While the aim of the model in the context of this paper is to replicate aspects of human reference resolution, which *per se* do not require particular assumptions about embodiment, real-time performance, and distributed, parallel processing, we believe that these three aspects will become critical in future extensions and most importantly in practical applications of the model (e.g., in situations where robots have to interact with humans in uncontrolled, natural environments.)[5]

4.1 Experiments

We conducted four experiments with the model (based on the experiments reported in (Tanenhaus et al., 1995; Eberhard et al., 1995)). As robotic agent we used an ActivMedia Pioneer Peoplebot robot equipped with an onboard PC104 board running LINUX, a Sony color camera on a pan-tilt unit, a BT848 video framegrabber, and SoundBlaster-compatible sound card. Specifically, the robot was placed about two yards in front of a whiteboard, on which colored squares where placed in two different arrangement according to the two-referent and one-referent condition depicted in Figure 3. For each condition, two sentences were read out loud to the model:

1. “Put the red square on the orange square on the blue square”
2. “Put the red square that is on the orange square on the blue square”

The predictions are (analogous to the human case) that the model should without difficulty be capable of processing the first sentence in the “two-referent condition” and the second sentence in both conditions, while it should fail to process the first sentence in the “one-referent condition” due to overspecification of the referent.

[5] We have conducted several preliminary experiments that vary some of the three dimensions in order to test the robustness of the model (e.g., different lighting conditions and speakers). Systematic experiments that establish the limits of perceptual and internal real-time processing are currently under way. These include, in particular, a richer conceptual processing basis together with an extended vocabulary.

4.2 The Model Implementation

The model was entirely implemented in the ADE system under development in our lab (Andronache & Scheutz, 2004, 2005) and distributed over four computers.[6] ADE is a JAVA-based agent development environment that builds on the general agent architecture framework APOC, which views agent architectures as networks of connected components with different levels of activation that communicate via four types of communication links (for more details on APOC, see Scheutz, 2004; Scheutz & Andronache, 2004). Since the focus in this paper is on the incremental construction of interpretations of sentences, we have modeled the semantic representation in great detail. For example, there are separate APOC components for conceptual frames corresponding to actions denoted by verbs, spatial relationships denoted prepositions, and individual terms corresponding to objects in the environment (such as the blocks). In contrast, we abstracted over details of sensory processes at the level of the APOC architecture and implemented all constituent parts in a single APOC component (i.e., the auditory and visual modules in one APOC component each). Moreover, for efficiency reasons, we have not attempted a one-to-one implementation of all functional modules of the model (in terms of individual APOC components), but rather merged functional modules whenever possible to obtain a better run-time performance, which is critical for real-time processing (e.g., the speech recognition and the color-shape detection modules are implemented together in one APOC components). It should be noted, however, that this is only an implementation choice (at the level of the APOC architecture) that does not change the functional architecture of the model. Figure 6 shows the mapping from the high-level functional architecture (in Figure 5) onto APOC components.

At the APOC level, the functional organziation of the implemented architecture consists of eight concurrently operating APOC components:

- The SPEECHINPUT component wraps around the Sphinx II decoder[7], which performs word recognition with Hidden-Markov Models based on the Fourier-transformed auditory signal. The SPEECHINPUT represents the *Speech Recognition* component of the architecture, providing lexical representations of spoken words to other APOC components.

- The WORDANDSYNTAXANALYSIS component receives data from the SPEECHINPUT component, analyzes the words and instantiates other components based on input. The speech input is then forwarded to the newly instantiated components. *Word Analysis* is implemented in this APOC component, which also performs some high-level *Syntax Analysis* (e.g., phrase structure identification associated with verbs).

- PUT components are representations of the verb "put" as architectural components and are therefore *Conceptual Frames*. Whenever a PUT component is created, it automatically instantiates a REFERENTANALYSIS component.

- REFERENTANALYSIS components perform some *Syntax Analysis/Lexicon* in that they incrementally parse descriptions trying to determine unique referents that correspond to them.

[6] The particular setup of the model reported here was demonstrated at the AAAI 2004 Intelligent Systems Demonstration (Scheutz, Andronache, & Eberhard, 2004). The specific details of the implementation of the model and the runtime environment ADE are described elsewhere (Scheutz & Andronache, n.d.).

[7] See HTTP://CMUSPHINX.SOURCEFORGE.NET/HTML/CMUSPHINX.PHP

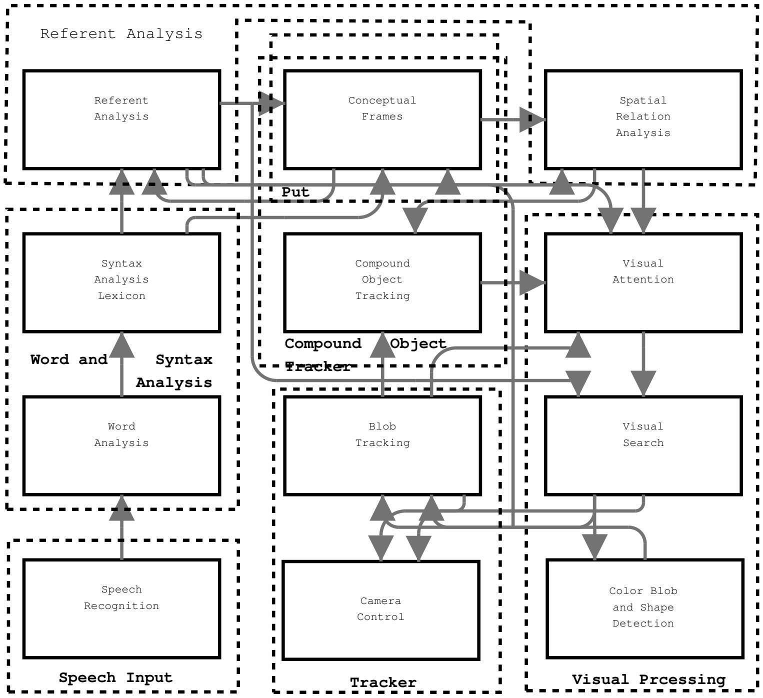


Figure 6: The mapping of high-level functional components onto APOC components indicated by dashed lines.

They also instantiate relational *Conceptual Frames* (such as ON components). Thus, REFERENTANALYSIS components perform the functions of *Referent Analysis* and *Spatial Relation Analysis*.

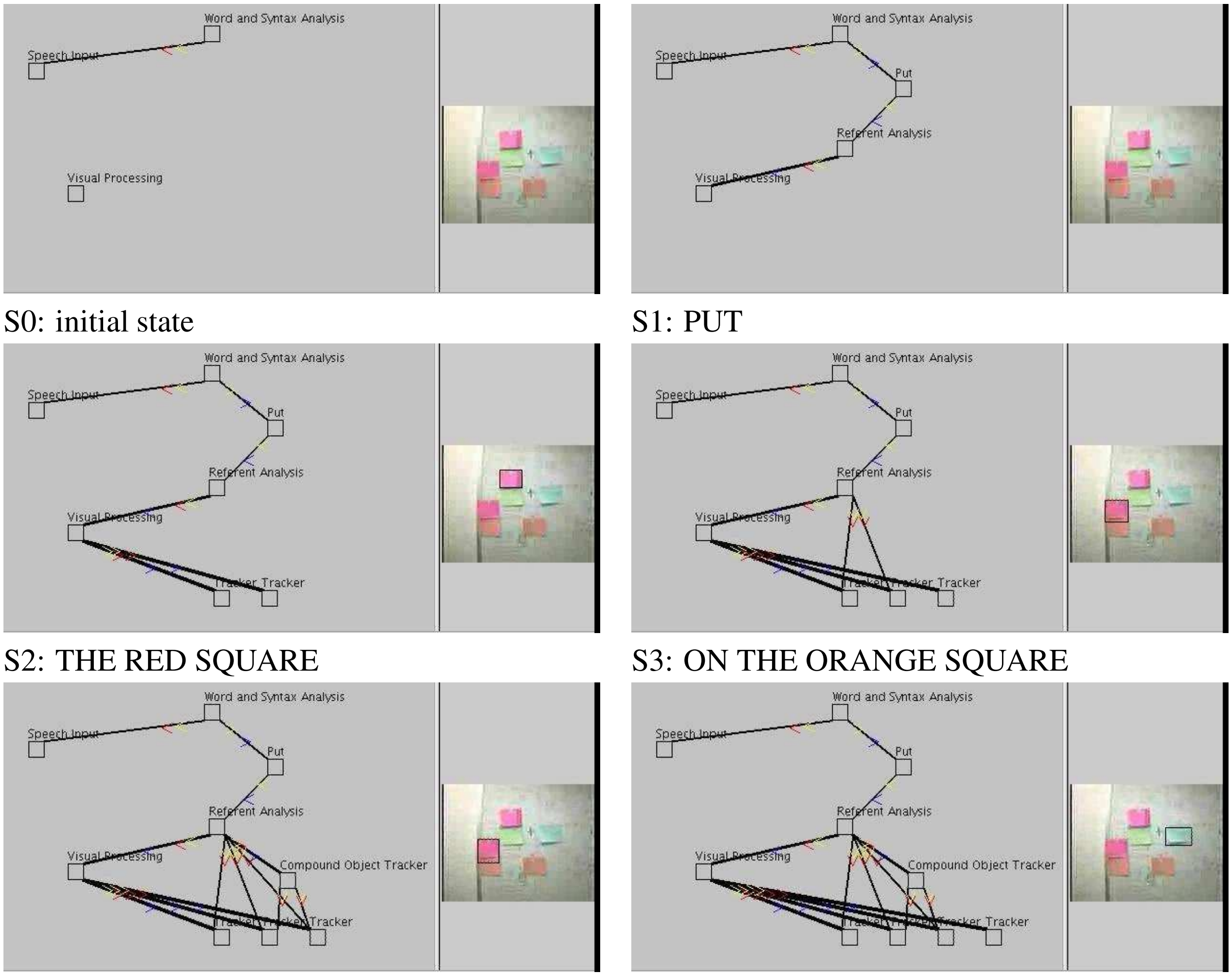
- The VISUALPROCESSING component combines several visual functions, which it can perform on the pixel image it obtains as input from the camera via the framegrabber. It also receives color and/or shape and position data from the REFERENTANALYSIS component. This data specifies the color currenly being sought and the region of the image in relation to objects of the previously sought color ('no relation' can be specified). This component corresponds to *Color Blob and Shape Detection*, *Visual Attention*, and *Visual Search* in the functional architecture.

- Each TRACKER component receives upon instantiation a blob from the VISUAL PROCESSING. Subsequently, the tracker receives information about all blobs of the same color and/or shape with the original blob and attempts to determine which of the current blobs corresponds to the tracked blob. A TRACKER can also send commands to the camera, although that feature was not used in this experiment. Thus, a TRACKER performs *Blob Tracking* and *Camera Control*.

- Each COMPOUNDOBJECTTRACKER component is connected to two other components (trackers or compound trackers, such as a "Left of"). The functions are performed by COMPOUNTOBJECTTRACKER components is *Compound Object Tracking*.

4.3 Results

In the following we report and discuss the results from all four experimental runs. For both conditions of sentence (1), we also show screen shots of the display of the running model (in Figure 4.3 and Figure 4.3, respectively). The left part of each figure depicts the state of the model after processing the phrase shown below the figure, the right part of each figure shows the image taken from the robot's camera at that point. Squares on the left with names on top depict active components in the running model that run in their own computational process (possibly on different machines). Links depict information flow. A black frame around a colored square on the right depicts the model's focus of attention (if attention is focused on a particular object).[8]



S0: initial state

S1: PUT

S2: THE RED SQUARE

S3: ON THE ORANGE SQUARE

S4:

S5: ON THE BLUE SQUARE

Figure 7: Various states of the model during the experimental run for sentence (1) in the two-referent condition.

We first consider six main states of the model in the two-referent condition with sentence (1)

[8] Since the arrangement of squares on the whiteboard is the same as the one depicted in Figure 3, the color of each object in the grayscale reproduction of the screen shots can be inferred from its relative location.

(corresponding to the screen shots in Figure 4.3):

S0 The SPEECHINPUT component is listening of auditory input, the VISUALPROCESSING component is acquiring images from the framegrabber at a rate of about 10 Hz. Focus of attention is the center of the image.

S1 The WORDANDSYNTAXANALYSIS component recognize the term "put" and instantiates a conceptual representation of "put". Speech input processing control is passed to the PUT component, which in turn creates a REFERENTANALYSIS component.

S2 The VISUALPROCESSING component searches the image for red squares. It finds two red squares in the image and thus instantiates two TRACKER components, one for each square, which are added the set of possible referent **PR**. The focus of attention is shifted to object closest to the previous focus of attention (i.e., the upper most red square).

S3 The VISUALPROCESSING component restricts its search area to those regions directly below the red objects previously identified and searches for yellow objects. A single object is identified, and a TRACKER component is created for that object, which causes the focus of attention to shift to the newly created tracker.

S4 A COMPOUNDOBJECTTRACKER component is instantiated and connected to the trackers of the respective red and orange squares. The set of possible referent **PR** is updated (i.e., the red square that is not on an orange square is removed as is does not meet the condition for membership at this point). Since **PR** only one object at this point, the REFERENTANALYSIS component indicates to the is returned to the PUT component, which after receiving "on" immediately returns control to the referent analysis component in order to determine a second unique object as the target of the "put" action.

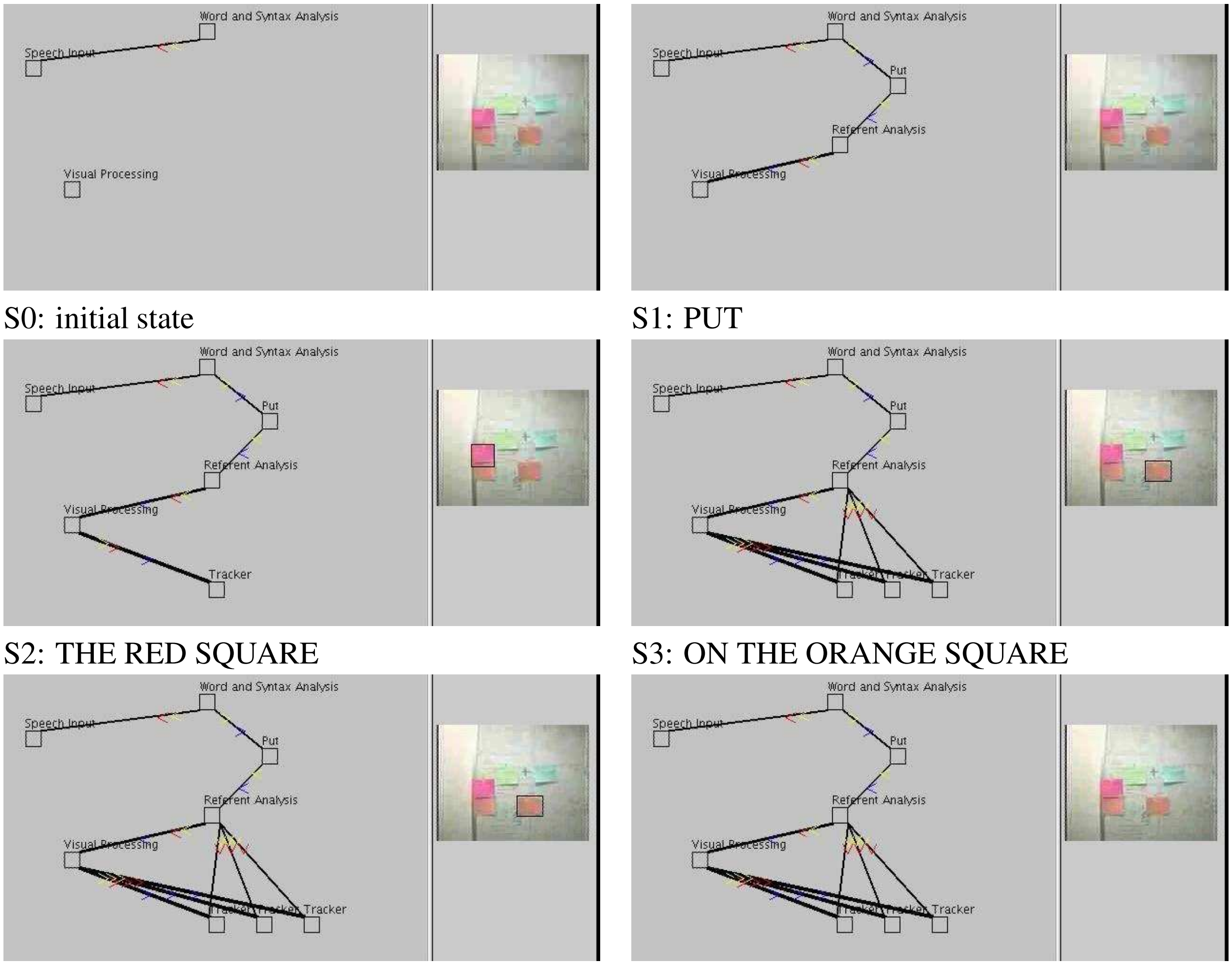
S5 A new, unrestricted visual search is started to find a blue object with empty space above (which is required for the target location of the "put" action).[9] Since there is only one blue object (with empty space on top) in the picture, the target location is identified, a new TRACKER component is instantiated, and the focus of attention is again shifted to the newly created tracker. At this point, speech input processing control is passed to the PUT component, which has finished processing its two constituent parts (the object of the "put" action and the target location). Hence, the semantic interpretation of the sentence is complete and control is returned to WORDANDSYNTAXANALYSIS component.

The model also exhibits the same sequence of states for sentence (2) in the two-referent condition, the only difference being that the REFERENTANALYSIS component also parses the interjected phrase "that is" in S3, which is simply taken to further restrict **PR**.

Now, consider the main states of the model in the one-referent condition (corresponding to the screenshots in Figure 4.3):

S0 Same as in two-referent condition.

[9] The constraint of space on "top" is passed from the PUT component, to the REFERENCEANALYSIS component, to the VISUALPROCESSING component.



S0: initial state

S1: PUT

S2: THE RED SQUARE

S3: ON THE ORANGE SQUARE

S4: ON

S5: THE BLUE SQUARE

Figure 8: Various states of the model during the experimental run for sentence (1) in the one-referent condition.

S1 Same as in two-referent condition.

S2 The VISUALPROCESSING component searches the image for red squares. It finds only one red square in the image and thus instantiates only one TRACKER component, which is added the set of possible referent **PR**. The focus of attention is shifted to object closest to the previous focus of attention (i.e., the upper most red square). Since **PR** has only one object at this point, control is returned to the PUT component, which after receiving "on" immediately returns control to the referent analysis component in order to determine a second unique object as the target of the "put" action.

S3 A new, unrestricted visual search is started to find an orange object with empty space above (which is required for the target location of the "put" action).[10] While two orange squares are found in the image, the constraint of having "empty space on top" eliminates one of them, hence no TRACKER component is instantiated for it. For the other orange square, a new TRACKER component is instantiated, and the focus of attention is again shifted to the newly created tracker. At this point, speech input processing control is passed to the PUT component, which has finished processing its two constituent parts (the object of the "put" action and the target location). Hence, the semantic interpretation of the sentence is complete and control is returned to WORDANDSYNTAXANALYSIS component.

S4 The WORDANDSYNTAXANALYSIS components receives the lexical input "on", which is taken to belong to a new sentence.[11]

S5 Since the model has only representations of sentence forms that start with verbs, the WORDANDSYNTAXANALYSIS cannot parse the prepositional phrase "on the blue block" and as a result the focus of attention is lost. Note that in an extended model, in which WORDANDSYNTAXANALYSIS component can deal with sentence forms that start with prepositional phrases, a new REFERENCEANALYSIS component would have been instantiated at this time together with a TRACKER component tracking the blue square, and control of the speech input would have been passed back to the WORDANDSYNTAXANALYSIS component since **PR** had only one element, the blue square.

If, however, the phrase "that is" is added (as in sentence (2)), then in states S3 through S5 will look exactly like in the 2-sentence condition as the WORDANDSYNTAXANALYSIS component will return speech input control to the REFERENTANALYSIS component upon receiving "that", which will the subordinate clause. Note that the result of processing the subordinate clause does not modify **PR** as **PR** had only one member to begin with. This is different from the above two-referent condition, where processing of the subordinate clauses does restrict **PR**.

In sum, the model exhibits the predicted outcome and, moreover, shows the same shift in focus of attention as found in the human experiments. The main difference between the model and the

[10] The constraint of space on "top" is passed from the PUT component, to the REFERENTANALYSIS component, to the VISUALPROCESSING component.

[11] Note that the model might deviate from the human case here in that the prosodic information provided by the speaker does not indicate the end of a sentence, while the semantic interpretation does. This conflict between semantic and prosodic (i.e., in this case syntactic) information might cause humans to reanalyze the interpretation of the sentence. The model, however, does not process and therefore does not take into account prosodic information.

human in the one-referent condition is that humans, by virtue of carrying out the "put" action, have to reach a decision of what to do, while the model does not actually perform any action (out of lack of appropriate effectors on the robot).

5 Discussion

The experimental results suggest that the model can capture important aspects of the processes involved in human reference resolution. Different from most cognitive models, which are simulated, the proposed model is intrinsically a robotic model that has to cope with real-world inputs with all the associated problems. Moreover, it processes information in real-time given the real-time nature of the auditory input. Both properties are important for applications of the model, for example, in mixed human-robot teams, where robots need to communicate with humans using natural language. Finally, the model implementation exhibits real parallel processing given that it is running on multiple computers. The last point is critical for extended versions of the model as the model's distributed nature makes it more likely that the model will "scale up" (i.e., will work the same way for much larger vocabularies and conceptual knowledge bases).

While there are some recent robotic models in AI to resolving references to objects in visual scences, which are related in spirit to the model proposed here, most notably Deb Roy's work (e.g., Roy, 2002; Roy, Gorniak, Mukherjee, & Juster, 2002), these approaches typically either employ parse trees and grammars (as in the Bishop system of Gorniak & Roy, 2004), or they are not aimed at trying to model human processes of incremental language processing (but rather at showing, for example, how word meanings emerge and can be used to refer to objects in a shared scence, e.g., Steels & Kaplan, 2002, or how reference can be resolved with minimal computational effort in a behavior-based robotic system , e.g., Horswill, 2001).

The model also illustrates other aspects of language understanding systems that are situated in an environment and have to interact in real-time. For example, it hints at ways of anchoring meanings of symbols through its interactions with the environment: primitive terms like "red" or "square" are *grounded* in perceptions by virtue of trackers that establish a causal loop with the objects in the world that have these visual properties. Hence, trackers can be viewed as *architectural representations* of the objects they track. Similarly, relation terms like "on" and "next-to" are grounded in complex trackers that do not track objects themselves, but the representations of those objects that are the relationship they are supposed to track. Consequently, instantiated complex trackers are *architectural representations* of tuple instances of relations, whereas their types at the architecture level correspond to the relations themselves. As such, complex trackers can be used both to represent facts ("A blue block is on a red block") as well as to ground the meaning of referential expressions with relation terms (e.g., "the blue block on the red block").[12]

On a speculative note, the model suggests that humans *in general and by default* may not obligatorily construct or use syntactic parse trees to establish reference. This claim does not imply that humans are incapable of constructing parse trees because evidence from the processing of 'garden path" sentences in the psycholinguistic literature clearly shows that they can. Our proposal

[12] Note that under this construal, perceptual facts are always positive, i.e., it is not possible to represent the "absence of a perception" such as that the blue block is *not* on the red block (which is in line with psychological assumptions about the nature of cognition, see, for example, Hearst, 1991).

is consistent with recent studies by Ferreira (Ferreira, Bailey, & Ferraro, 2002; Ferreira, 2003) showing that comprehenders use heuristics to arrive at a "good enough" semantic interpretation (see also Sanford & Sturt, 2002). Our model suggests that parse trees may be used by higher-level supervisory control to reanalyze sentences in which the initial interpretation fails.

The current implementation of the model has several limitations, the most obvious one of which is its minimal vocabulary and conceptual knowledge (e.g., it consists of a few color terms with their associated Gaussian distributions in RGB space, some preposition terms together with the procedural knowledge about their geometric meaning, and a handful of verbs together with their phrase structure). Moreover, since thematic roles are currently only assigned in the context of commands (where the agent is implicitly taken to be the listener and does not require explicit representation), the current implementation of the model cannot parse other sentence types without a modification of the WORDANDSYNTAXANALYSIS, which allows it to instantiate a REFERENTANALYSIS component before instantiating a conceptual frame.[13] Also, the model currently does not employ any learning, hence cannot answer any questions about how humans can learn how to resolve references from experience. Rather, the core assumptions giving rise to the reference resolution algorithm in Section 3.3 are hard-coded into the model. While the model was not intended to be a model of "learning to resolve reference", it would still be interesting to see what the necessary architectural constraints are that would allow the model to learn the proposed algorithm (e.g., as a natural step in the process of acquiring compositional semantics, cp. to Schoenemann, 1999). Finally, the model only incorporates automatic processing mechanisms, which in humans are entrained through repeated language usage. While these mechanisms can to some extent recover from errors as could be seen from the examples with "that"-clauses, where control was returned to the REFERENTANALYSIS component even after a unique referent had already been determined for further referent parsing, the model cannot deal with and recover from many other parsing errors that humans can handle. Extensions to the model will be needed to account for the various ways in which humans reanalyze sentences syntactically (with or without explicit knowledge grammar rules) in order to find alternative semantic interpretations, possibly involving supervisory control.

6 Conclusion

In this paper we have proposed an model for incremental resolutions in humans, which is built on and thus incorporates as part of its processing architecture the pragmatic assumption about human communication we called *Minimal Information Processing Principle*: Speakers aim to provide the minimal amount of information necessary to distinguish their intended referent(s) from alternatives in the domain of reference that is shared with their listeners. By the same token, listeners expect speakers to adhere to this principle. Violations of this principle, which may be in the form of underspecification (ambiguous reference) or overspecification (superfluous reference), can cause confusion or comprehension difficulty as demonstrated by the eye-tracking experiments discussed in the introduction.

The model showed the same performance as humans, suggesting that it was capable of captur-

[13] We are currently testing a preliminary version of an augmented WORDANDSYNTAXANALYSIS. Note, however, that this modification does not require changes to the REFERENTANALYSIS component, which implements the incremental resolution of reference.

ing some important aspects of human incremental language processing. It can, therefore, not only be of use for further cognitive modeling, but also as an alternative to standard language processing approaches in AI. For one, language processing in the model takes place in parallel and the semantic interpretation of the meaning of the sentence is completed when the last lexical item is processed, which facilitates real-time processing as the complexity of reference resolution is linear in the length of the phrase. This is different from many AI systems, which build explicit parse trees as part of their interpretation of a sentence, where the effort can be exponential in the worst case. The current implementation of the model has its limitations (such as the limited vocabulary and conceptual knowledge) that need to be overcome before it can be applied in a practical system. However, we believe that its functional architecture provides the foundations for very general language processing systems that interact with humans via language and therefore will need to adhere to the pragmatic assumptions underlying natural language discourse.

References

Andronache, V., & Scheutz, M. (2004). ADE - a tool for the development of distributed architectures for virtual and robotic agents. In *Proceedings of the fourth international symposium "from agent theory to agent implementation"*.

Andronache, V., & Scheutz, M. (2005). ADE - an architecture development environment for virtual and robotic agents. *International Journal of Artificial Intelligence Tools*, forthcoming.

Chambers, C. G., Tanenhaus, M. K., Eberhard, K. M., Filip, H., & Carlson, G. N. (2002). Circumscribing referential domains in real-time language comprehension. *Journal of Memory and Language*, *47*(1), 30-49.

Chomsky, N. (1957). *Syntactic structures*. The Hague: Mouton.

Chomsky, N. (1965). *Aspects of the theory of syntax*. Cambridge, MA: MIT Press.

Chomsky, N. (1986). *Knowledge of language*. New York: Praeger.

Clark, H. H. (1996). *Using language*. Cambridge: Cambridge University Press.

Clifton, C., Speer, S., & Abney, S. P. (1991). Parsing arguments: Phrase structure and argument structure as determinants of initial parsing decisions. *Journal of Memory and Language*, *30*(2), 251-271.

Eberhard, K. M., Spivey-Knowlton, M. J., Sedivy, J. C., & Tanenhaus, M. K. (1995). Eye movements as a window into real-time spoken language comprehension in natural contexts. *Journal of Psycholinguistic Research*, *24*, 409-436.

Ferreira, F. (2003). The misinterpretation of noncanonical sentences. *Cognitive Psychology*, *47*(2), 164-203.

Ferreira, F., Bailey, K., & Ferraro, V. (2002). Good-enough representations in language comprehension. *Current Directions in Psychological Science*, *11*(1), 11-15.

Ferreira, F., & Clifton, C. (1986). The independence of syntactic processing. *Journal of Memory and Language*, *25*, 348-368.

Frazier, L. (1995). Constraint satisfaction as a theory of sentence processing. *Journal of Psycholinguistic Research*, *24*, 437-468.

Frazier, L., & Fodor, J. D. (1978). The sausage machine: A new two-stage parsing model. *Cognition*, *6*, 291-325.

Gorniak, P., & Roy, D. (2004). Grounded semantic composition for visual scenes. *Journal of Artificial Intelligence Research*, *21*, 429-470.

Grice, H. P. (1975). Logic and conversation. In P. Cole & J. Morgan (Eds.), *Speech acts* (pp. 41–58). New York: Academic Press.

Hearst, E. (1991). Psychology and nothing. *American Scientist*, *79*, 432-443.

Horswill, I. (2001). Tagged behavior-based architectures: Integrating cognition with embodied activity. *IEEE Intelligent Systems*, *September/October 2001*, 30-38.

Kamide, Y., Altmann, G. T. M., & Haywood, S. L. (2003). The time-course of prediction in incremental sentence processing: Evidence from anticipatory eye movements. *Journal of Memory and Language*, *49*(1), 133-156.

Karttunen, L., & Peters, S. (1975). Conventional implicatures of montague grammar. *Berkeley Linguistic Society*, *1*, 266-278.

Kimball, J. (1973). Seven principles of surface structure parsing in natural language. *Cognition*, *2*(15-47).

MacDonald, M. C., Pearlmutter, N. J., & Seidenberg, M. S. (1994). Lexical nature of syntactic ambiguity resolution. *Psychological Review*, *101*(4), 676-703.

McClelland, J. L., & Rumelhart, D. E. (1988). *Parallel distributed processing* (Vol. 1 and 2). Cambridge: MIT Press.

Roy, D. (2002). Learning visually-grounded words and syntax for a scene description task. *Computer Speech and Language*, *16*(3).

Roy, D., Gorniak, P., Mukherjee, N., & Juster, J. (2002). A trainable spoken language understanding system. In *Proceedings of the international conference of spoken language processing.*

Sanford, A. J., & Sturt, P. (2002). Depth of processing in language comprehension: not noticing the evidence. *Trends in Cognitive Sciences*, *6*(9), 382-386.

Scheutz, M. (2004). Apoc - an architecture for the analysis and design of complex agents. In D. Davis (Ed.), *Visions of mind* (p. forthcoming). Idea Group Inc.

Scheutz, M., & Andronache, V. (n.d.). A distributed realtime architecture for incremental language processing.

Scheutz, M., & Andronache, V. (2004). Architectural mechanisms for dynamic changes of behavior selection strategies in behavior-based systems. *IEEE Transactions of System, Man, and Cybernetics Part B*, forthcoming.

Scheutz, M., Andronache, V., & Eberhard, K. (2004). A robotic model of human reference resolution. In *Proceedings of aaai 2004* (p. 2 pages).

Schoenemann, P. T. (1999). Syntax as an emergent characteristic of the evolution of semantic complexity. *Minds and Machine*, *9*(3), 309-346.

Sedivy, J. C. (2002). Invoking discourse-based contrast sets and resolving syntactic ambiguities. *Journal of Memory and Language*, *46*(2), 341-370.

Sperber, D., & Wilson, D. (1995). *Relevance: Communication and cognition* (2nd ed.). Oxford: Blackwell Publishers Inc.

Spivey, M. J., Tanenhaus, M. K., Eberhard, K. M., & Sedivy, J. C. (2002). Eye movements and spoken language comprehension: Effects of visual context on syntactic ambiguity resolution. *Cognitive Psychology*, *45*(4), 447-481.

Stalnaker, R. C. (1978). Assertion. In P. Cole (Ed.), *Syntax and semantics, vol. 9: Pragmatics* (p. 315-332). New York: Academic Press.

Steedman, M. J. (1989). Grammar, interpretation, and processing from the lexicon. In W. Marslen-Wilson (Ed.), *Lexical representation and process* (p. 463-504). Cambridge, MA: MIT Press.

Steels, L., & Kaplan, F. (2002). Bootstrapping grounded word semantics. In E. J. Briscoe (Ed.), *Linguistic evolution through language acquisition: formal and computational models* (p. 53-74). Cambridge, UK: Cambridge University Press.

Tanenhaus, M. K., Spivey-Knowlton, M. J., Eberhard, K. M., & Sedivy, J. C. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science*, *268*, 1632-1634.

Tanenhaus, M. K., & Trueswell, J. C. (1995). Sentence comprehension. In J. L. Miller & P. D. Eimas (Eds.), *Handbook of perception and cognition. vol 11: Speech, language, and communication.* (p. 217-262). Orlando: Academic Press.

Winograd, T. (1972). *Understanding natural language*. New York: Academic Press, Inc.

Winograd, T. (1973). A procedural model of language understanding. In R. Schank & K. Colby (Eds.), *Computational models of thought and language.* W. H. Freeman and Company.