Timing of visual and spoken input in robot instructions.

Joerg Wolf and Guido Bugmann
Robotic Intelligence Laboratory, School of Computing, Communications and Electronics,
University of Plymouth, Drake Circus, Plymouth PL4 8AA, United Kingdom.
joerg.wolf@plymouth.ac.uk, gbugmann@plymouth.ac.uk

Abstract:
Trainable robots will need to understand instructions by humans who combine speech and gesture. This paper reports on the analysis of speech and gesture events in a corpus of human-to-human instructions of the dealing phase of a card game. Such instructions constitute an almost uninterrupted stream of words and gestures. One the task of a multimodal robot interface is to determine which gesture is to be paired with which utterance. The analysis of timing of events in the corpus shows that gestures can start at various time relatively to the speech, from 5 seconds before speech starts to 4 seconds after speech ends. The end of a gesture never precedes the corresponding utterance. A simple algorithm based on temporal proximity allows to pair correctly 83% of gestures with their corresponding utterances. This indicates that timing carries significant information for pairing. For practical applications, however, more reliable pairing algorithms are needed. The paper also describes how individual actions can be grouped into a gesture and discusses the integration of semantic information from gesture and speech.

Keywords: human-computer interaction, natural language understanding, multimodal interfaces, service robots, speech events timing.

1. Introduction

Future service robots can not be completely pre-programmed by the manufacturer. There are far too many possible tasks and user-dependent variants. These robots will need to learn interactively from their users. They will need to be *programmable by anybody* (naive users / without training) and not just by engineers, roboticists and computer scientists. A user-programmable robot thus requires an interface that is natural to the user. One approach to the design of a truly easy-to-use interface is by examining the interaction between people. By observing instructions from human teachers to human students, guidance is sought here for the design of a robot acting as the student. In a previous project in our laboratory on Instruction Based Learning project (IBL) (Kyriacou, 2004; Bugmann *et. al.* 2004) it was shown that through the analysis of the teacher's utterances, it is possible to:

- Identify primitive procedures that the robot has to be able to carry out (the robot's "prior knowledge")
- Write and tune speech-recognition software to address and combine these primitive procedures.

This approach to the definition of the robot's functionality and natural-language interface (NLI) has been described as "corpus-based robotics" (Bugmann et. al. 2004, Bugmann et al., 2005).

In the current "multimodal IBL" project (MIBL), the analysis of human-to-human multimodal instructions combining gesture and voice is explored. The test case is that of a human explaining to another human how to play a specific card game. The teacher and students communicate using voice and card manipulations on touch screens. Such actions could theoretically also be detected with a vision system and most of the results presented here are hopefully valid for multi-modal systems using vision-based gesture recognition. Previous reports on multimodal input systems using touch screens, such as (e.g. Boves et al., 2004) impose strict constraints on when inputs can be provided and note that "subjects hardly ever combined pen and speech". In the free-flowing instruction application described here, there is frequent combination of speech and gesture and there is a need for assessing which gesture corresponds to which speech act. Other works, e.g. (Perzanowski et al., 1998) and subsequent works e.g. (Perzanowski et al., 2003), do not provide details on how elements of a stream of words and a stream of gestures are associated.

In section 2, the experimental setup and corpus collection procedure are summarized. In section 3, the methods of gesture recognition and categorization are described. In section 4, timing data are shown that suggest a method for synchronizing visual input (gestures) and auditory input (speech recognition). In section 5, the semantic information provided by gesture is discussed. Section 6 offers concluding comments.

2. Experimental setup and corpus.

The MIBL project is focused on the generation of programs for the robot from multimodal instructions. To reduce to a minimum the problems of visual perception and manipulation of real-world objects, it was decided to use a touch screen as interface between the human and the robot, in addition to a voice interface. The screen represents the world as the robot would see it through it's vision system. The user is able to point to and manipulate objects on the screen as a demonstration on how to do the task. At the same time the user gives verbal instructions. Touch-screens have been used in multimodal human-robot interfaces for different applications, for example by (Perzanowski *et al.*, 2001), or for investigations in human communication (De Ruiter *et al.*, 2003)

A great advantage of using a screen representing the robot's world is that the learning robot to be designed can be simulated, while the interaction and interface to the robot does not change from the one used to collect human data. It also allows focusing research on human-robot interfaces without the need of having to build a robot. Details on how subjects were organized into teachers and students, and the experimental protocol can be found in (Wolf and Bugmann, 2005). In short, subjects where initially students and became teachers in later sessions. 21 teaching sessions were recorded. Two of them concerned the training of the initial teachers and are not used for system development. Ten sessions are used for system development (training set) and nine sessions will be used for system testing (test set). The corpus comprises video recordings of the sessions (see figure 1), recordings of the voices of the teacher and the student, and recordings of movement of cards on the screen. For each teacher-student pair, the instructions session and two following games were recorded. This project focuses on analysing the teacher's speech and gestures, in order to build a robot able to understand human instructions.

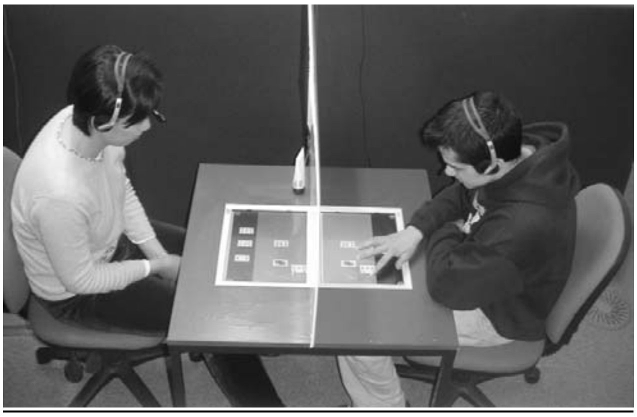

Figure 1: Corpus collection setup. The instructor on the right moves a card on the touch screen. The learner sees a copy of the move on her screen. The separation panel between instructor and learner force the gesture component of the communication to take place via the touch screen and can easily be recorded. Each screen has a small "private" black band representing the hand of the player. The larger green area represents the table and is shared by the two players.

A transcription tool (Wolf and Bugmann, 2005) was designed that allows producing XML files including gesture and speech act timings. The entries on gestures were generated automatically using a recognition method described in the next section. The transcription of speech was done manually by adding speech tags to the gesture tags. The analysis of the corpus is not complete yet. This paper reports on data covering only the initial phase of the game instruction: how to deal cards.

3. Gesture recognition

In card games, gestures can be pointing gestures, gestures moving cards from one place to another (e.g. stack to table, hand to table), re-arranging gestures (making a group of cards look tidier) and turning over gestures. A touch screen operates as an additional mouse to a computer. The effect of a user touching the screen is signalled as a mouse button-down event. Moving cards on the touch screen is intuitively done by touching the card and dragging it to another position. The resulting data is a trail of X, Y coordinates of where the card is going. In case of a real service robot, this tracking data of cards on the screen could be the output the service robot's vision system.
The "analogue" trail of X, Y data of a cards position is then registered as a movement from a start area to a destination areas e.g. move(pile, temp1). The areas numbers and their boundaries are defined from observations of where the movements of the players usually end, namely: stockpile, table, hand1, hand2, temp1 and temp2 (figure 2). The stockpile's position is set by the system.

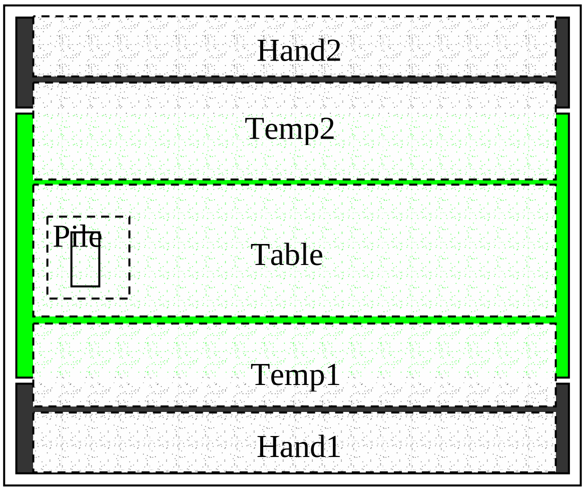

Figure 2. Areas defined on the touch screens. Temp1 and Hand1 are on the teacher's side. Temp2 and Hand2 are on the student's side. The teacher and the student can only see and manipulate cards in their hand area.

The categorization of areas is a straight forward comparison between coordinates and area boundaries. An analogue belief system, see Roy, D. (2005), could be used instead in a real service robot, if the vision system output coordinates are noisy or if there are no clear cut boundaries. In our experiment it was found that a statistical categorization is not required, in other cards games however, the situation can be more complicated. Within a move of a single card, users sometimes stop and then continue to move the same object until it reaches its final location. This is meant to be a single move by the user, but how can the robot recognize that? The strategy used here is to wait until the human picks up another object, which automatically implies that he has finished with the previous one. This is generally true with a touch screen, where there is only a single mouse cursor. With real vision, a better method might be to use a timeout.

Gestures which the same start and destination position are pointing gestures. Gestures with the same start and destination area are re-arranging gestures. Gestures with different start and destination areas are card displacements. Gestures are recorded in a transcription file in XML format including time of start and end of movement, player doing the move, card identity and start position and destination, such as for instance:

```
<objmove t="2416" user="t" ID="D/5" from="+Table+Stock" to="+Temp2" until="2442"></objmove>
```

4. Synchronizing visual and auditory input.

In the transcriptions of the human-to-human dialogues, utterances and gestures are grouped together by the operator doing the transcription. This provides reference data, in order to design a system that can automatically group utterances with gestures. The challenge here is to pair automatically the correct gestures with the correct utterances. In the domain of card games we found that most often a single utterance U_1 was associated with several actions $G_1, G_2, G_3, ...$ forming a group of actions. Groups of actions, such as dealing 3 cards, are characterized by short time delay separating individual actions. Figure 3A shows that most actions in a group are separated by less than 2 seconds. However, the time intervals between groups are very short too, and time intervals between groups (end to start) are often smaller than 2 seconds (Figure 3B). Therefore, groups can not reliably be identified on the basis of time intervals. A safer and reliable method is to group actions according to their start and destination areas and to the type of action performed on the cards (see e.g. table 1). The end of a group of actions is identified either from the start of a new group of actions, or from a time-out of 2 seconds (from figure 3A).

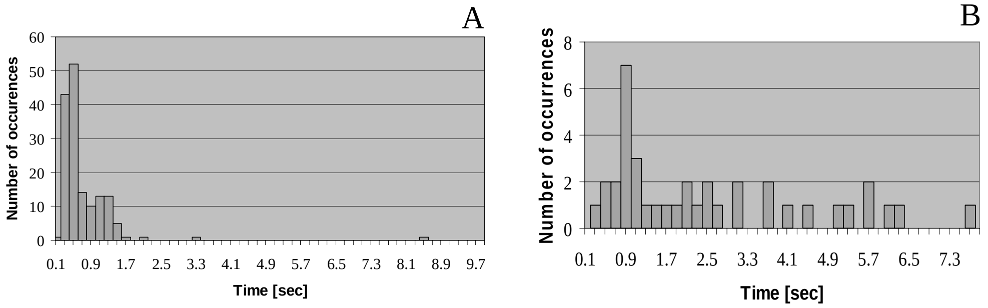

Figure 3. A) Histogram of time intervals between individual actions in a group of actions. B) Histogram of time intervals between action groups (gestures).

The second question is how groups of actions (which we will call a “gesture”) can be associated with the corresponding utterances. The speech recognition engine (NUANCE 8.5) provides the times of start of speech and the times of end of speech. The same information is provided for gestures using the method cited above. Thus, we explored the possibility to associate gestures and utterances by comparing their relative timings. Gestures tend to start either before or after speech in equal proportion (Fig. 4A). However, gesture rarely starts after speech ends (Fig. 4B). Figures 4A and 4B show that the first action in a group usually occurs later than 5.5 seconds before the start of speech and not later than 4 seconds after the end of speech. This time “buffer” at each end of the utterance (see figure 5) can unfortunately not often be used to assign a start of gesture to its utterance. The reason is that the time interval between utterances (end of the previous to the start of the next) is generally smaller than 9.5 second (4 + 5.5) (Fig. 4C) and buffers belonging to different utterances generally overlap. Observation of the instruction process shows that the teacher never pauses for a long time, producing a nearly continuous flow of words and actions. This is especially true of instructions of the dealing phase. Thus, the time periods where buffers do not overlap (e.g. period B in figure 5) are relatively short. Only 40% of gestures start during this non-overlapping period and can be unambiguously paired. The start times of the remainder of the gestures fall into area A (figure 5) where pairing is uncertain. More elaborate rules of pairing are required. For instance, additional information may be obtained from the timing of the end of the gesture. One observation might be of interest. Figure 4D shows that the end of a gesture always occurs at least 1.6 second after start of speech. In other words, subjects sometime start the gesture well before speech, but always start speaking before the paired gesture is completed. Unfortunately the reverse is not true. In several cases, subjects started a new utterance before the gesture related to the previous one had ended. Therefore, the timing of end of gestures does not immediately appear to be helpful to decide the pairing between gesture and utterance. To assess if relative timing carries useful information at all, we attempted to assign gestures starting in period A (figure 5) to the nearest utterance. This resulted in a total of 83% correct pairings (including the 40% correctly paired in time period B). This shows that timing contains significant information exploitable for pairing. However this result is of little practical use, as pairing errors are bound to cause serious problems in instruction understanding. What is needed is a pairing system that either produces a safe pairing or signals its uncertainty.

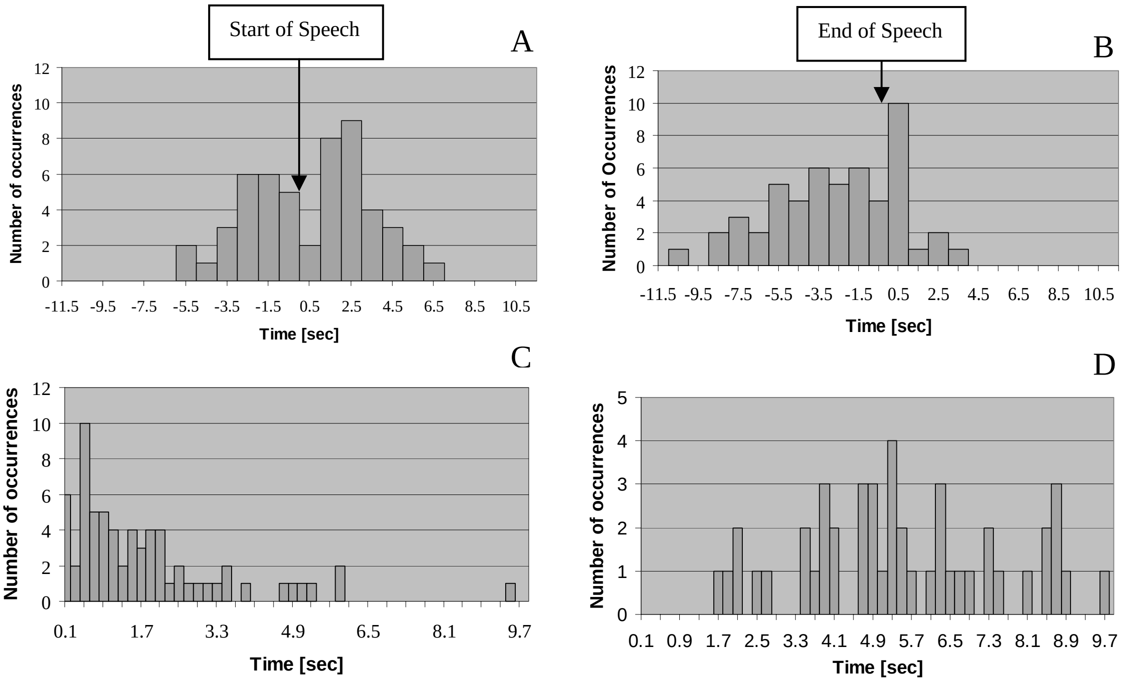

Figure 4: A) Histogram of the time intervals between start of speech and start of the first action in a group. B) Histogram if time intervals between the end of speech and the start of the first action in a group of actions. Only groups of actions associated with the speech event are plotted. C) Histogram of time-intervals between speech events (end of previous to start of next). D) Histogram of time difference between end of the last action in a group and the start of speech.

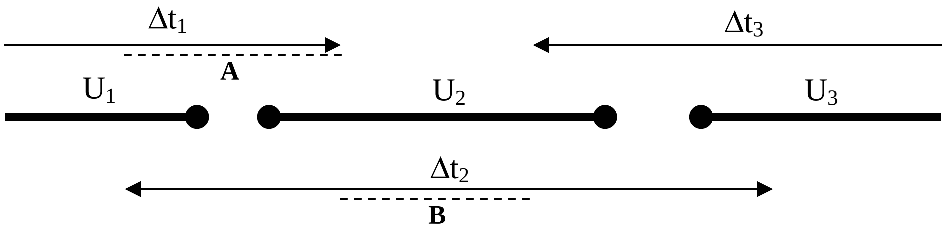

Figure 5: Illustration of overlapping time windows. Three utterances U1, U2 and U3 define each a time widow Dt1, Dt2 and Dt3 which extend the utterance duration by 5 sec before its start and 4 sec after its end. During the time span A, overlapping time windows make it impossible to assign gesture to either U1 or U2. Only during time span B can a gesture reliably be assigned to U2, based on the start time of the gesture.

5. Semantic representation and role of gesture

Card games typically consist of three phases: dealing, playing and a post-game phase where players count their points. In this paper we focus on the explanations covering the dealing phase. Table 1 shows an example of how verbal instructions and gesture are combined in such explanations. Dealing explanations comprise a sequence of actions, in contrasts to play instructions which include mainly rules (not shown in this paper). In the card game Scopa, which was used during corpus collection, every player gets three cards and another four cards are dealt face up on to the table. All subjects described dealing as a sequence of six steps (see e.g. table 1). In the MIBL corpus it was found that all teaching proceeds via spoken instructions accompanied by simultaneous execution, making descriptions of actions much more detailed, and easier to understand for a human or robotic student. The role of gesture is often to specify spatial coordinates of verbal instructions (e.g. instruction 2) or to resolve references such as “these” (e.g. instruction 4), as also noted by other authors (e.g. Perzanowski et al., 2000).

*Do as I do***.** A curious problem with task instructions in unconstrained spoken language is that the speaker uses interchangeably “I”, “you” and “we”. This appear to come from the fact that the teacher is demonstrating and expects the robot to copy his/her behaviour in most cases. To some extent, the availability of an example to follow appears to require less linguistic rigour from the teacher. It is likely that instructions given over the phone would be much more precise. A formal linguistic analysis is bound to meet serious problems here, but this is not the topic of this paper. It appears that, at least in the dealing case, it could be a good strategy to ignore most of the speech and learn to mirror the teacher’s actions. Only is cases where cards are invisible to the student (in area hand1) would speech processing provide necessary information. Practical implementation will verify if this is possible. Indeed, in explanation of the rules of the game speech cannot be ignored. Further analysis of the corpus will clarify this.

6 Concluding comments

The presented data show that gestures can start before, during or after an utterance within limited time windows, but never end before the utterance starts. Simple pairing rules based on parts of these data produce correct pairings for 83% of gestures. This is encouraging given the complexity of the situation analyzed here, characterized by a free flow of gesture and verbal instructions. For practical applications however, different characteristics are demanded from a pairing algorithm. It must either given a correct pairing, or signal it inability to provide a pairing. In the latter case appropriate repair dialogues can then be initiated. Another issue to be considered is the fact that dialogues with robotic systems are likely to exhibit different timing characteristics than the ones between humans. These may show a simplification of the pairing problem. Otherwise, and depending on the causes of the difficulty, one may have to introduce timing constraints on the sequence of speech/gesture through dialogue strategies. Another avenue is to exploit the semantic analysis of the speech to identify paired gesture.

Once utterances and gestures are paired, they can provide complementary information, as identified in other works. However the quality of speech in terms of grammatical rigour appears to be very poor here, certainly poorer that in the IBL corpus where gestures were not allowed. In this case, it is possible that this bears no consequence, as it may turn out that most learning can be done by imitation. In task instruction, there is a fixed amount of information to communicate and user may just “spread” that information across modalities. Thus, multimodal communication may not always carry more information than unimodal information.

7 References

L. Boves, A. Neumann, L. Vuurpijl, L. ten Bosch, S. Rossignol, R. Engel, and N. Pfleger. (2004) Multimodal Interaction in Architectural Design Applications. Proceedings, UI4ALL 2004: 8th ERCIM Workshop on "User Interfaces for All" 28-29 June 2004, Vienna, Austria.

Bugmann G., Klein E., Lauria S. and Kyriacou T. (2004) Corpus-Based Robotics: A Route Instruction Example. in Proceedings of IAS-8, 10-13 March 2004, Amsterdam, pp. 96-103.

Bugmann G., Wolf J. C., Robinson P. (2005) The Impact of Spoken Interfaces on the Design of Service Robots. Industrial Robot, 32:6, 499-504

Roy, D. (2005), "Semiotic Schemas: A framework for Grounding Language in Action and Perception", Elsevier, Artificial Intelligence Journal, Volume 167, Issues 1-2, Pages 170-205 (http://web.media.mit.edu/~dkroy/papers/pdf/aij_current.pdf)

Mellish, C.S., (1985), "Computer interpretation of natural Language descriptions", Ellis Horwood Limited, Chichester, ISBN 0-85312-828-6

Dennis Perzanowski, William Adams, Alan C. Schultz and Elaine Marsh (2000) Towards Seamless Integration in a Multi-modal Interface. Proceedings of the Workshop on Interactive Robotics and Entertainment, Carnegie Mellon University: AAAI Press, 3-9, April 2000.

Perzanowski, D. , Shultz A.C. and Adams W. (1998) , "Integrating Natural Language and Gesture in a Robotics Domain" Proceedings of the IEEE International Symposium on Intelligent Control: ISIC/CIRA/ISAS Joint Conference, Gaithersburg, MD: National Institute of Standards and Technology, 247-252, September 1998.

Perzanowski, D., Brock, D., Blisard, S., Adams, W., Bugajska, M., Schultz, A., Trafton, G., Skubic, M. (2003) Finding the FOO: A Pilot Study for a Multimedia Interface, In Proceedings of the IEEE Systems, Man, and Cybernetics Conference, Washington, DC

Schank, R. and Abelson, R., (1977), "Scripts Plans Goals and Understanding", Book published by Lawrence Erlbaum Associates Inc, New Jersey, U.S.A., ISBN 0-470-99033-3

Wolf J.C., Bugmann G. (2005) Multimodal Corpus Collection for the Design of User-Programmable Robots. Proc. Taros'05, London, p. 251-255.

Table 1. Example explanations of the dealing phase. The table shows utterances spoken by the teacher, related gestures and meaning of the combined input. Some details are discussed in the text. Times are given in 1/10th of a second.

No	Text – 03	Gestures	Semantics
1	Ill just explain how you deal the cards <tv t="2396" until="2428">		gamestate = learn_dealing
2	er what you do is first of all is	<objmove t="2416" user="t" ID="D/5" from="+Table+Stock" to="+Temp2" until="2442”/>	start_learn_sequence(seq1)
	er you take three cards for Yourself	<objmove t="2446" user="t" ID="C/2" from="+Table+Stock" to="+Temp2" until="2460"/>	set1 = D/5,C/2,H/QQ and owner(set1,robot)
	<tv t="2431" until="2477">	<objmove t="2463" user="t" ID="H/QQ" from="+Table+Stock" to="+Temp2" until="2477"/>	seq1_step1 = goal(move set1 from stock to temp2)
3	face down and ill take three	<objmove t="2482" user="t" ID="D/QQ" from="+Table+Stock" to="+Temp1" until="2496"/>	set2 = D/QQ,D/KK,D/AA
	<tv t="2486" until="2513">	<objmove t="2499" user="t" ID="D/KK" from="+Table+Stock" to="+Temp1" until="2510"/>	seq1_step2 = goal(move set2 from stock to temp1)
		<objmove t="2512" user="t" ID="D/AA" from="+Table+Stock" to="+Temp1" until="2522"/>	owner(set2,human)
4	you take these into your black	<objmove t="2531" user="t" ID="D/QQ" from="+Temp1" to="+Hand1" until="2544"/>	Ref. Resolution: “these” = set1
	Area	<objmove t="2547" user="t" ID="D/KK" from="+Temp1" to="+Hand1" until="2567">	seq1_step3 = goal(move set1 from temp2 to hand2)
	<tv t="2540" until="2563">	<objmove t="2570" user="t" ID="D/AA" from="+Temp1" to="+Hand1" until="2577">	
5	so you can drag them down		Ref. Resolution: “them” = set1 , hence “down”=hand2
	<tv t="2564" until="2575">		no action required, already done
6	and then er turn them over	<objrot t="2599" user="t" ID="D/AA" roty="0" />	Ref. Resolution: “them” = set1
	<tv t="2606" until="2627">	<objrot t="2644" user="t" ID="D/QQ" roty="0" />	seq1_step4 = goal(turnover set1)
		<objrot t="2613" user="t" ID="D/KK" roty="0" />	
7	so you can see them and obviously i cant see them <tv t="2660" until="2675">		Ref. Resolution: “them” = set1 (this sentence can be used for confirmation)
8		<objmove t="2580" user="t" ID="D/AA" from="+Hand1" to="+Hand1" until="2582"/>	The card is not mentioned and didn’t change location. Therefore this move is unimportant
9	and then what we do next is er <tv t="2680" until="2704">		- (supports we are still in seq1)
10	put four cards face up on the	<objmove t="2695" user="t" ID="H/2" from="+Table+Stock" to="+Table" until="2715"/>	set3 = H/2,D/3,C/KK,D/JJ
	Table	<objmove t="2719" user="t" ID="D/3" from="+Table+Stock" to="+Table" until="2736"/>	seq1_step5 = goal(set3 cards from pile to table)
	<tv t="2708" until="2760">	<objmove t="2740" user="t" ID="C/KK" from="+Table+Stock" to="+Table" until="2753"/>	
		<objmove t="2668" user="t" ID="D/JJ" from="+Table+Stock" to="+Table" until="2692"/>	
11	so ill just turn those over	<objrot t="2808" user="t" ID="D/JJ" roty="0" />	Ref. Resolution: “those” = set3
	<tv t="2821" until="2834">	<objrot t="2820" user="t" ID="H/2" roty="0" />	seq1_step6 = goal(turnover set3)
		<objrot t="2832" user="t" ID="D/3" roty="0" />	
		<objrot t="2844" user="t" ID="C/KK" roty="0" />	