Combining Planning with Gaze for Online Human Intention Recognition

Socially Interactive Agents Track

Ronal Singh, Tim Miller, Joshua Newn, Liz Sonenberg, Eduardo Velloso, Frank Vetere
School of Computing and Information Systems, The University of Melbourne
{rr.singh, tmiller, newnj, l.sonenberg, evelloso, f.vetere}@unimelb.edu.au

ABSTRACT

Intention recognition is the process of using behavioural cues to infer an agent's goals or future behaviour. People use many behavioural cues to infer others' intentions, such as deliberative actions, facial expressions, eye gaze, and gestures. In artificial intelligence, two approaches for intention recognition, among others, are gaze-based and model-based intention recognition. Approaches in the former class use gaze to determine which parts of a space a person looks at more often to infer a person's intention. Approaches in the latter use models of possible future behaviour to rate intentions as more likely if they are a better 'fit' to observed actions. In this paper, we propose a novel model of human intention recognition that combines gaze and model-based approaches for online human intention recognition. Gaze data is used to build probability distributions over a set of possible intentions, which are then used as priors in a model-based intention recognition algorithm. In human-behavioural experiments ($n = 20$) involving a multi-player board game, we found that adding gaze-based priors to model-based intention recognition more accurately determined intentions ($p < 0.01$), determined those intentions earlier ($p < 0.01$), and at no additional cost; all compared to a model-based-only approach.

KEYWORDS

Intention Recognition; Gaze; Planning

ACM Reference Format:
Ronal Singh, Tim Miller, Joshua Newn, Liz Sonenberg, Eduardo Velloso, Frank Vetere. 2018. Combining Planning with Gaze for Online Human Intention Recognition. In *Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), Stockholm, Sweden, July 10–15, 2018,* IFAAMAS, 9 pages.

1 INTRODUCTION

Our visual behaviour is intrinsically linked to how we plan and execute actions. Our eyes are constantly scanning and interrogating the environment around us in a continuous cycle of observation and prediction [9]. As a consequence, by monitoring the eye movement behaviour of others, we are able to infer their future actions. This ability was crucial to our success as a species—the *cooperative eye* hypothesis suggests that the unusually large visible white area in our eyes (the sclera) evolved due to the social requirements of tasks requiring joint attention, as it enabled us signal to other humans where we were looking at [15, 27]. Animals that evolved in competitive environments have their sclerae hidden (e.g. dogs) or pigmented (e.g. gorillas). An understanding of this behaviour can provide us with opportunities in both cooperative and non-cooperative settings. For example, a poker player observing which cards their opponents have been gazing upon to potentially determine which hand they are trying to compile; or an assistant handing a surgeon a tool that they will potentially use next. Despite the large role that gaze plays in social interactions and in action execution, it has received little attention for these tasks in artificial intelligence. Social artificial agents may be able to improve their interactions with humans by anticipating the human's intentions by combining the human's gaze and ontic behaviours. Agents can then adapt their behaviour in light of these intentions or devise plans to provide proactive support to the human counterparts. In this paper, we propose an intention recognition approach that incorporates visual behaviour into model-based intention recognition using automated planning and demonstrate how it substantially improves the recognition performance.

With decreasing cost and increasing robustness, eye trackers are entering the consumer market, particularly as game controllers [28]. Whereas most modern applications involve explicit control (e.g. clicking or typing with the eyes), there is a huge potential for intelligent user interfaces that use gaze implicitly to derive people's intentions and adapt the interaction accordingly. Although the relationship between gaze and action is well-understood, we still lack computational models that integrate eye tracking data into predictive systems. Existing work has demonstrated that it is possible to classify in which activity a human is currently engaging based solely on eye movement data [5] and to predict intention using machine learning algorithms such as *support vector machines* [2, 13] or *decision trees* [14]. However, in our applications of interest, we do not have sufficient data to train such models, so instead, we opt for a model-based approach to intention recognition.

Recent work has successfully used automated planning for model-based intention recognition of intelligent agents [23, 24], and some preliminary experiments show its potential for human intention recognition in simple tasks such as shape drawing [29]. In these approaches, for each potential intention, the recognition approach uses an automated planner to generate two plans: (1) a possible future trajectory that achieves the intention while corresponding to some sequence of already performed actions; and (2) the optimal plan for this intention that does not correspond to the already performed actions. Plans that are closer to the optimally-computed plans are deemed as more likely to be a rational 'fit' for the intention, and therefore more likely.

We hypothesise that incorporating gaze data to form a prior probability of these intentions can improve prediction accuracy.

Proc. of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2018), M. Dastani, G. Sukthankar, E. André, S. Koenig (eds.), July 10–15, 2018, Stockholm, Sweden.

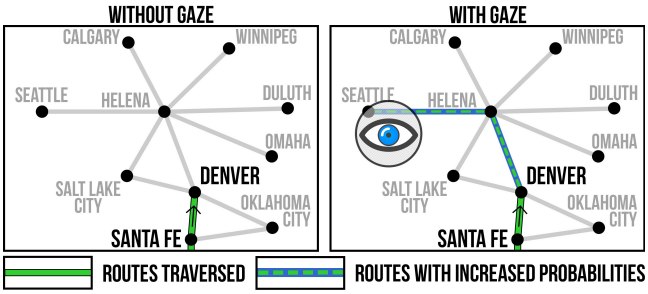

Figure 1: Intent Recognition with Gaze Scenario

We use automated planning techniques to generate candidate plans. The candidate plans and the actions of the user are then combined with the user's gaze distribution to infer their possible intentions.

Figure 1 shows a simple example of our idea. In this example, based on the board game *Ticket to Ride* described in Section 4, a person is trying to navigate a path (e.g. direct a vehicle) between Santa Fe, and one of the other cities in the graph. The intention recognition problem is to determine the destination city. On the left, we see that the route from Santa Fe to Denver has already been traversed. We argue that this implies that the probability of the final destination being Oklahoma is smaller than that of any other of the other nodes: a rational navigator would more likely traverse the path from Santa Fe to Oklahoma directly. Existing model-based approaches would rate this as such. However, from this single traversed route, we are unable to distinguish the probability of the outer nodes (Seattle, Calgary, Winnipeg, etc.)—they are all the same distance from Denver. However, consider the example on the right, in which we know that the person has been looking at the route from Helena to Seattle. We argue that now, this represents a potential future action and that Seattle is a more likely final destination than Calgary, Winnipeg etc. We argue further that Calgary, etc., are still more likely than Oklahoma, which fits neither our observed navigation actions nor our observed gaze actions.

The key contribution of this work is the prediction of human intent using a novel combination of model-based and gaze-based approaches. We define a model of intention recognition using gaze and combine this with other model-based approaches. We evaluate this approach on twenty people playing a digital multi-agent board game while their gaze data was being recorded. We compared a model-based approach using only game-state data (*Model Only*) to the same approach enhanced with gaze-based priors (*Gaze+Model*). Although the model-based approach was reasonably successful at predicting intentions, our results showed that our enhanced model: (1) more accurately predicted the future intentions of players; (2) was able to make these predictions earlier in the game; and (3) was able to calculate these predictions with no additional computational execution cost. These contributions are significant because they provide an empirical groundwork for designing predictive gaze-based systems.

2 RELATED WORK

2.1 Eye Tracking in HCI

Current interactive applications that employ affordable eye trackers fall into two main areas: *gaze interaction* and *context-aware computing*. Gaze interaction remains an ongoing area explored by HCI researchers as a form of input, particularly for selection, through novel interaction techniques (e.g. [6, 8]). Till date, the most significant uptake of eye tracking technology has been in the gaming industry with Tobii Gaming[1] leading its charge by providing access to advance affordable eye trackers for use in over 70+ gaze-enabled games. Gaming has long been a popular application domain for researchers to explore novel inputs and gaze is no exception. A survey on gaze interaction in games by Velloso and Carter [28] provides an overview of how gaze has been incorporated into games through a taxonomy of gaze-based game mechanics. While most mechanics involve explicit real-time interaction (e.g. aiming & shooting), implicit interaction using adaptive AI was found to be highly relevant to our work, where the game AI would learn from the player's visual attention patterns to predict future actions. For example, Munoz et al. predicted player's actions in Super Mario Bros by analysing intentions from gaze data using an artificial neural network [18]. Similarly, Hillaire et al. used simple prediction models to determine the direction in which players were going to turn in a 3D environment based on gaze behaviour [11]. Likewise, gaming has been pushing AI applications such as by using probabilistic networks in a game to predict the player's next move to precompute graphics or by generating intelligent behaviours in non-player characters (NPC). For instance, Wetzel et al.'s AI game engine dynamically adapts its strategy based on the combination of eye movements and player's actions [30].

Alternatively, eye tracking presents opportunities for systems to be aware of its users' activities (e.g. reading, browsing watching videos) through the implicit monitoring of eye movements ([5, 16]). For instance, Kunze et al. found that reading on different document types can be recognised automatically from reading behaviour (74% success rate) [16]; demonstrating its potential to perform activity recognition at a finer level. While a majority of studies in both areas have shown promising results over the years, they primarily focus on the on the present (what the user is current doing), or in the short term (what the user is going to do next). Our interest ultimately lies in recognising what a user might do much further into the future (i.e. intentions) in order to build systems that can make adaptions in a timely fashion.

2.2 Intention Recognition via Gaze

Here, we summarise existing works that focus specifically on performing intention recognition via gaze which typically use a machine learning approach. Huang et al. investigated the predictive role of gaze using in a collaborative task where participants role-played a sandwich making scenario between a worker and a customer [13]. Using the ingredients eventually chosen as the ground truth for the customer's intentions, they measured the extent to which gaze cues served as predictors of their choices. They attempted two approaches: (1) a simple attention-based intention predictor that performed predictions according to which ingredient the customer most recently fixated upon, and (2) by using a SVM-based classifier using four gaze features (number of glances, duration of first glance, total duration of glances, and whether a particular ingredient was most recently glanced at. The first approach outperformed random guesses achieving an estimated accuracy of

[1] http://tobiigaming.com

65 percent while the SVM classifier achieved an estimated real-time accuracy of 83 percent. Similarly, Bednarik et al. trained an SVM classifier using a number of gaze features to identify the type of task a user was performing when playing the 8-tiles puzzle game. The types of tasks included planning, where the participant identifies the possible actions to take, and cognition, where participant identifies the particular information the participant is currently processing [2]. The gaze features were either *fixation-based* (e.g. fixation length, fixation count), *saccade-based* (e.g. count, area, duration), and *interface-based* (e.g. total no. of visited areas).

Other approaches have also been used. Ishii et al. proposed an algorithm based on *decision trees* that estimated the user's conversational engagement with an artificial salesperson using attributes beyond gaze direction alone [14]. They trained a decision tree to serve as their engagement estimation model using four attributes, i.e. gaze direction transition, transition duration, amount of eye movement and pupil size). Their model could predict the user's disengagement with an accuracy of around 70 percent. Similar to Huang et al., Andris et al. also used a sandwich making scenario but used a different approach to modelling prediction and intention i.e. Epistemic Network Analysis (ENA) [1]. Their analysis gives an overall picture of the unfolding gaze patterns in dyadic collaborations. A recent study by Newn et al. used a different approach by employing human subjects to infer intention using gaze visualisations of a player over a strategic turn-based game called *Ticket to Ride* [20, 21]. Their study demonstrated the predictive ability of gaze and its potential to reveal plans early if gaze is displayed from the beginning of the game when no actions have yet been played.

2.3 Model-Based Intention Recognition

An alternative way to perform intention recognition is to use a model of the possible behaviours to perform a forward projection of candidate plans, starting from a sequence of recently-observed behaviours, and to identify those plans that best fit the observations. Most existing works in this area use *plan libraries* for candidate plans, and assess how likely it is that the observed sequence of behaviours is a good fit for the prefix of such a plan. Many models of plan libraries have been investigated, such as hidden Markov models [3] or belief networks [12]. Sukthankar et al. [25] survey existing approaches.

A downside of using plan libraries is that they require a set of potential plans for the other agent; which may be both prohibitively large, but also inflexible if the observed agent deviates from this set. Alternative approaches use concepts such as action models to temporally project possible agent plans into the future, essentially generating a set of candidate plans at runtime. Such models use belief-desire-intention plans [26] or planning models [24, 29].

The work that we build on in this paper is on intention recognition as *planning*. Here, the set of candidate plans is not generated *a priori*, but instead, a planning tool is used to generate the set of possible plans that correspond to the observed prefix, and to assess their likelihood.

Ramirez and Geffner [24] solve this problem by generating two plans for each possible intention: the optimal plan for the intention in which the observed sequences of actions are satisfied by the plan; and the optimal plan for intention in which the sequences are not satisfied. They then determine the probability of seeing these observations for the intention by measuring the *difference* between the two plans, and using Bayes rule to determine the probability of the goals, given some prior distribution over the goals.

More recent work simplifies this approach to reduce its computational burden. Vered and Kaminka [29] extend Ramirez and Geffner [24] for *online* intention recognition. Instead of generating an optimal plan that does not satisfy the observations at each time step, they generate a single optimal plan for each intention, and then at each time step, generate a plan that satisfies the observations, and take the *ratio* between the cost of the original non-satisfying plan with the satisfying plans as a score, normalising these scores to get a final probability distribution. Masters and Sardina [17] independently make a similar observation for intention recognition in path planning, using the cost difference between the optimal plan that satisfies the observations and the optimal plan in general, thus also avoiding calculation of plans that do not satisfy observations. However, they show that for path planning (not task planning), this can be calculated without considering the observations at all. They prove that the ranking of goals can be achieved by knowing only the agent's start point, potential goals, and its current location.

In our current and planned projects, our applications typically require a form of planning, so we use a model-based approach to recognise intent from gaze behaviour. This is because of two reasons: (1) humans make plans within the context given therefore giving us a set of possible plans and (2) as gaze is linked to action—observing the gaze of a human can give an indication of their intentions. In the next section, we describe a model that integrates gaze to perform human intention recognition.

3 MODEL

In this paper, we propose a system that consists of two independent components that form the input of our intention recognition algorithm: (1) the *gaze model* that processes the gaze information and uses the concepts of fixation count and fixation length to determine the probabilities of different intentions; and (2) the *planning-based model*, which takes an action model and an observed sequence of actions, and determines the probability of intentions based on how well they fit the observed sequences.

3.1 Problem Formulation

We ground our problem on similar definitions based on intention recognition as planning [29]. Informally, there is a set of possible intentions that the observed agent can achieve, a set of actions that can be used to achieve it, and a set of observations we receive. The intention recognition problem is to determine the likelihood of the different intentions given the observations.

In this context, observations are divided into two categories: (1) *ontic actions*; and (2) *gaze actions*. Ontic actions are those in which the agent under observation modifies the (physical or virtual) world. Gaze observations are those in which the agent under observation is looking at 'regions' of the world, in which the regions are related to ontic actions. For example, in the navigation scenario, gaze actions are those that look at the different routes and destinations, while moving between cities are ontic actions.

Formally, an online goal recognition problem R is a tuple $R = \langle W, s_0, I, A, G, O_a, O_g \rangle$. W is the world in which the agent operates, $s_0 \in W$ is the initial state of the observations, I is the possible set of

goals/intentions, A is the set of ontic actions available to achieve the intentions, G is the set of possible gaze actions, O_a is the sequence of ontic action observations (made up of actions from A), and O_g is the set of gaze observations (made up of observations from G).

This model aligns with the model of intentions outlined by Pacherie [22], who argues that there are three types of intentions: (1) *distal intentions* (D-intentions), which are the long-term intentions of the agent and when achieved, terminates the agent's practical reasoning process in relation to the current actively (generally); (2) *proximal intentions* (P-intentions), which are short-term intentions that an agent derived to help achieve D-intentions; and (3) *motor intentions* (M-intentions), which are the grounded motor actions used to achieve D-intentions. In this work, we are not considered with M-intentions, however, we note that this model aligns with the notions of D-intentions and P-intentions: D-intentions are the set of potential *intentions* I, and P-intentions are the actions A. As such, the set of intentions in our work is the set $\Xi = I \cup A$. That is, an intention is either an intention to achieve an end state or an intention to execute some action in the future. This aligns with Bratman's [4] notion that '*intentions that*' the world is in a particular state (a goal) expresses an intention around the world, while '*intentions to*' perform an action express an intention about an action.

A *solution* to the problem R is a probability distribution over the set of possible intentions Ξ, indicating the likelihood of each being real, based on the observations O_a and O_g. For the remainder of the paper, when we discuss intentions, we mean the set Ξ. From a modelling perspective, we can treat all intentions as actions, and have dummy terminal actions for each intention in I that correspond to that state being reached.

In this model, we assume a STRIPS-like model of ontic actions [10], with each action consisting of a unique name, preconditions that determine under which worlds in W they can be executed, and effects that describe the changes made to W. Gaze actions have a *fixation count* that defines the number of times this gaze action we performed, and *fixation length*, which defines the time for which this gaze action was performed.

The computational problem is to determine how well the possible trajectories possible by sequences actions from A correspond to the observations seen so far (O_a and O_g). Assuming that the actor aims to behave rationally, then intuitively, the similarity between the observations and the prefixes of the trajectories represents the likelihood of taking those trajectories—a rational agent will prefer cheaper plans over more expensive plans. From their P-intentions, we can infer the likelihood of their D-intentions. In this paper, we build on the related work to combine it with gaze data with the aim of improving the accuracy of the predicted intentions.

3.2 Intention Recognition

Our intention recognition approach is divided into three steps, according to the different types of action observations. First, the gaze observations are used to determine the probabilities of different intentions computed from the fixation length and fixation count. Second, independently of the second step, a model-based intention recognition approach is used to determine the probability of observing future trajectories of actions based on the past observations. Finally, these two are combined to give final probabilities over the possible intentions. This separation of the two approaches supports online intention recognition that can incorporate new gaze data observations as they become available, without having to perform the expensive planning process again.

3.2.1 Combining Gaze and Planning. We first present the final step of our approach, which forms the backbone of the model. In an environment in which both gaze and action are possible actions, we hypothesise that we can combine the two for better intention recognition. Our first observation is that the two types of observations can be treated independently, primarily because they describe two different types of action.

Gaze actions are telling us about the possible future intentions that human under observation is considering. Provided the person is not attempting to deceive the opponent using their visual behaviour, they will typically look at the regions that they are considering for future action more than those that they have already executed.

For example, consider our earlier navigation example in Figure 1. The navigator would be more likely to cast gaze at regions in which they are considering as possible destinations. Further, when they perform the ontic action of traversing a route between two neighbouring cities, we hypothesise that they are in fact highly likely to look at that region immediately prior to the traversal. That is, they do not navigate blindly.

On the other hand, the ontic action observations are used to describe intentions *that have already been fulfilled*. Despite this, they can still be used to predict future actions. As we saw in our example, if the goal is to traverse entire paths, traversing a particular route indicates an intention that this will form part of a larger path, thus increasing the likelihood of the routes following the traversed route. To combine these two, we propose a simple model that treats the two types of observations independently initially and then combines them. However, the gaze and ontic actions are not entirely independent of each other: both are driven by the D-intentions. We model the gaze observations as priors over the chosen trajectories. The rationale behind this is clear: (1) a person is unlikely to act in any environment without sensing it first, thus gaze data related to particular intentions will almost always occur before any ontic actions; and (2) a person is less likely to look at parts of the environment with actions that they have already performed (unless that action can be performed again). As such, gaze actions provide a good guide to future ontic actions.

Formally, we can describe the problem as trying to estimate $P(i \mid O_a, O_g)$ for each $i \in \Xi$; that is, the probability of an intention i given ontic and gaze action observations O_a and O_g. In this model, we assume that gaze actions from O_g are excluded once the related intention is achieved, thus implying that the probability of O_a is not influenced by the probability of O_g; although the inverse is not the case — past actions influence where people gaze. With this assumption, we can rewrite $P(i \mid O_a, O_g)$ as follows:

$$\begin{aligned} P(i \mid O_a, O_g) &= P(O_g, O_a \mid i) \cdot P(i) \quad \text{(Bayes rule)} \\ &= P(O_g \mid i) \cdot P(O_a \mid O_g, i) \cdot P(i) \\ &= P(O_g \mid i) \cdot P(O_a \mid i) \cdot P(i) \quad \text{(Assumption above)} \\ &= P(O_a \mid i) \cdot P(i \mid O_g) \quad \text{(Bayes rule)} \end{aligned} \tag{1}$$

Thus, the probability of a (P- or D-) intention i, given observed gaze and ontic actions O_g and O_a respectively, is the probability that the observed ontic actions would be taken if i was an intention, with

the prior that i is an intention from the gaze data. Given this model, the remainder of the problem is how to determine $P(O_a \mid i)$ and $P(i \mid O_g)$.

For our gaze model, we use fixation length and fixation count due to these two features being the basic features used in previous studies highlighted in Section 2.2. We define a simple model that uses these two measures to define the probability of each intention being a true intention. Recall from Section 3.1 that the gaze-based actions are summarised using fixation count and fixation length. A fixation is detected by the system when a human subject fixates on a target area for more than a threshold, for example, 60ms. A single fixation starts when the subject starts looking at a particular target (item of interest) and then ends when the subject looks at another target. The change of the target areas is detected by tracking the *(x,y)* coordinates collected by the eye tracker. For each fixation, we maintain a fixation length, which is the difference between the start and end times of each fixation. For each target area, we maintain the total number of times a user visits the target area and we refer to these counts as fixation counts. To cater for blinks (for periods shorter than 60ms), we merge the fixation lengths if the target being looked at remains the same after and before the blink.

The model of our data is assumed to be as follows. For each intention i, we have a pair: $\langle count_i, length_{i,j} \rangle$, in which $count_i$ represents how many times the gaze action corresponding to intention i was observed, and $length_{i,j}$ is a vector of j variables that represent the length of each gaze action. That is, $length_{i,j}$ represents the length of the j^{th} observation of i ($1 \leq j \leq count_i$).

We define fs_i, the *fixation score* for intention i as a weighted measure between fixation length and count:

$$fs_i = log(\lambda \cdot total\_time_i + (1 - \lambda) \cdot count_i) \tag{2}$$

where $total\_time_i$ is the sum of the fixation lengths for the intention i ($\sum_{i=1}^{i \leq count_i} length_{i,j}$), $\lambda \in [0, 1]$ is the relative weight given to the fixation lengths over fixation count. Because these are in different units (time and count respectively), it is likely that $total\_time\_i$ will be much higher than $count\_i$. In practice, we have defined $count\_i$ as the number of times i is looked at multiplied by the fixation threshold (e.g. 60ms), so the two variables are on the same 'scale'.

Note the use of the *log* function in Equation 2. Using fixation length and count directly results in the linear function that increases monotonically. However, people's intentions change, and they adopt new intentions over time. There are at least two options to mitigate this problem. First, discount the importance of earlier observations using a discount factor; or second, limit the increase in the importance of target areas as fixation length and count increase. By using the *log* function, we adopt the latter approach, which highlights important intentions while giving priority or importance to new intentions. We choose this approach because a person may stop looking at potential intentions once concrete plans have been formed, so discounting earlier intentions could miss important intentions.

Using the fixation weights, we define the probability of an intention i as its fixation weight normalised against other intentions:

$$P(i \mid O_g) = \frac{fs_i}{\sum_{j \in \Xi} fs_j} \tag{3}$$

where O_g is the set of fixation weights for the intentions representing the human gazing at targets that signal the intentions, and w_i is the fixation score for intention i.

3.2.2 Model-based Intention Recognition. Our model-based intention recognition is a simple generalisation of existing approaches of model-based intention recognition using planning. First, we define π_i as a plan that achieves intention i optimally while folding in observations O_a. That is, π_i is a sequence of actions from A, with prefix O_a, that executes P-intention i or achieves D-intention i, and where 'optimality' is defined by some criteria, such as length or cost of the trajectory. Note that there could be several such plans. Further, we define Π_i^{opt} as the *set* of optimal plans that execute/achieve i *without* (necessarily) folding in observations O_a.

We can now define the probability of observing O_a if intention i is a given trajectory that achieves i:

$$P(O_a \mid i) = \kappa \cdot \min_{\pi_i^o \in \Pi_i^{opt}} costdiff(\pi_i, \pi_i^o) \tag{4}$$

in which κ is a normalising constant and *costdiff* is a function that evaluates the 'cost' difference between two plans.

Informally, this equation states that the probability of seeing O_a if i is the intention is proportional to the cost difference between π_i and the closest optimal plans. Thus, a plan π_i that is closer to an optimal plan is more likely to have intention i. This definition may appear strange because O_a is a prefix of π_i by definition (π_i folds in observations O_a). However, we are not capturing the probability of O_a given just the plan π_i, but that the *intention* of π_i is to achieve i. Therefore, when assessing π_i, if π_i is far from the optimal plan, then the person's true intention is probably not i.

We do not provide a complete definition of *costdiff* here; however, several such measures are possible, such as the simple difference in length, ratio of lengths, or the difference between the specific actions in the plans. There are a number of such functions proposed by other researchers. See for example, [17, 24, 29]. In Section 4, we use a domain-specific definition, based on ideas from Ramirez and Geffner and [24] and Vered and Kaminka [29].

3.2.3 Implementation. To implement such a model requires us to calculate $P(i \mid O_a, O_g)$ for every potential intention. This can be computationally expensive.

Although our model combines the gaze and planning-based approaches into a single calculation (Equation 1), we note that a step-wise approach that first estimates $(i \mid O_g)$ from gaze actions can have practical value in cases where we may only care about the higher probability intentions. Planning-based intention recognition is likely to be (relatively) more expensive than our gaze model because we must perform an expensive planning step for each intention, whereas the gaze calculation is performed as a simple weighted sum. To mitigate this problem, a simple step would be to use the planning-based model only on those intentions that have a high probability from the gaze-based model; for example, the top N intentions, or intentions with a probability above some threshold.

Equation 4 uses the set of all optimal plans Π_i^{opt} to evaluate how likely it is to see observations for a given intention. This step is clearly computationally expensive for any non-trivial systems. As such, approximations that, for example, compute only one optimal

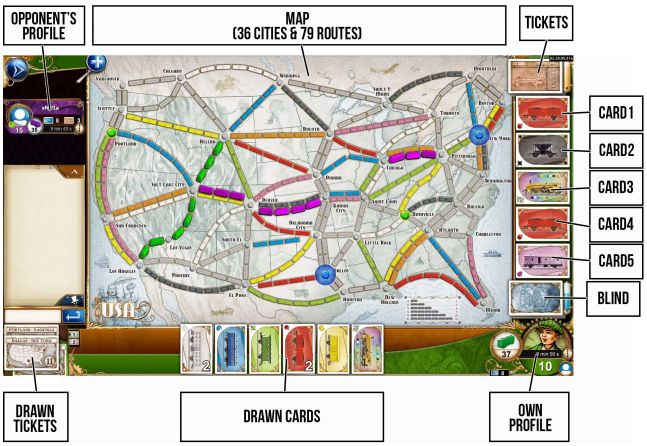

Figure 2: Top: A *Ticket to Ride* game instance with two players with respective target areas shown.

plan, may suffice in many settings. Alternatively, one could try to find the optimal plan that prioritises actions in π_i.

4 STUDY

For our study we use a multi-player game called *Ticket to Ride*[2], also used by Newn et al. [19, 20]. Figure 2 shows a screenshot of the game and its corresponding target areas. In this game, players compete to build train routes between adjacent cities (routes) across a map of North America, and gain points for building a connection between specific pairs of cities that they are assigned on ticket cards at the start of the game (e.g. from Dallas to New York), which form their top-level goals and are unknown to their opponent. Players receive two ticket cards at the beginning of the game. Players obtain the corresponding points stated on the card if they successfully connect them and will lose the same number of points if they do not. Players can draw additional ticket cards during their turn in the game if desired. Players can block each other's paths, as only one player can claim each train route. Therefore, players must plan their routes carefully to minimise the risk that an opponent will guess their intentions and block them by claiming the routes that they need first. Keeping information hidden is, therefore, core to the game, as a player can gain a significant advantage by correctly guessing their opponents' hidden objectives. These compelling reasons are why we selected this game for our study.

4.1 Data Collection

The study was conducted in a university usability lab, with two players in each session playing in separate observation rooms to simulate an online gameplay of *Ticket to Ride*. Both computer monitors (1920x1080) were fitted with a Tobii 4C eye tracker (*90 Hz*). We log the gaze of both players for each round played using a custom networked system for evaluating our intention recognition approach. Further, the system displayed a semi-transparent dynamic real-time heatmap for intention prediction (building on the findings of Newn et al. [20]) of one player to another in real-time to simulate an agent that reacts to the player's eye movements and elicit more realistic visual attention behaviours. The player who was given the ability to see gaze, the 'aware player', was shown the gaze visualisation of the 'naive player' throughout the game. The naive player was not informed that their gaze could be seen by their opponent, which made this a *keyhole plan recognition* scenario for the aware player [7]. We recorded the computer screens for the duration of each session.

First, both players together were given an initial briefing that explained that their gaze would be tracked for later analysis. Players were then allocated randomly to the two observation rooms, with one researcher in each, and were given a written overview of the study, consent form and basic demographic questionnaire. Following this, players were calibrated using default calibration procedure of the eye tracking device and instructed to play the game's interactive tutorial until they were satisfied that they understood the game. Players then proceed to play a normal game against each other to reduce any potential learning effects. In the next game, players played the enhanced version of the game where the aware player could see the gaze visualisation of the naive player. In total, we recruited twenty player pairs (22F/18M) from the same university, aged between 18 and 39 years ($M = 23.2$). We compensated players with a $20 gift card for their time. As we are only interested in the gaze of the 'naive player', we extracted the video recordings and the gaze logs (*x coordinate, y coordinate, timestamp*) for that round. We then encoded the game states of each game. For each player, we extracted: *cards drawn, routes claimed, scores* and *remaining resources*. In total we had around 421 minutes of game data with an average of 21 minutes/game (5 minutes).

4.2 Gaze Features and Planning

In this section, we briefly discuss some of the domain-specific details of our implementation.

Interface-based features: We divided the TTR game window into target areas as shown in Figure 2. The target areas signal an agent's intention. For example, if a player is looking at the cards, they may have formed the intention of picking up those cards; if a player is looking at a route, the player may have formed the intention of claiming that route. By considering the duration of the fixation and the number of times a player looks at a particular target, we make inferences about the likelihood of that intention being carried out. To establish which target was gazed at, we automated the mapping of x and y coordinates captured by the eye tracker to one of the target areas. The game state data, for example, the routes claimed by the players was manually coded.

Intention Model: We treated each route as a proximal intention. Using this, we were able to construct a modified planning problem that consisted of only the routes that had been looked at, along with their corresponding probability. This represents the priors for the planning algorithm. So, in this case, not only did we use the priors as probabilities, but also to reduce the size of the search problem.

Plan Generation: For each distal intention, the planner generated four different trajectories: two that fold in the observations and two that did not, corresponding to π_i and π_i^o respectively in Equation 4. We assume that there would be two strategies for fulfilling distal intentions: (1) build the shortest route; or (2) minimise the number of moves taken to build a route (that is, also including picking up cards, etc.). Thus, two plans that fold-in observations are generated — one for each possible way of achieving the distal

[2]http://www.daysofwonder.com/tickettoride/en/

intention — and two that do not — again, one for each possible way of achieving the distal intention.

For simplicity, we do not consider any other distal intentions that are possible in the game, such as claiming the longest route (which gains points in the game), trying to block other opponents routes, or actively trying to deceive other players on the intended route. The two former are the most common in our experience of playing and observing the game.

To further simplify the process of planning, we assume that the naive player always has wild cards. This means that we do not reason explicitly about the different cards the naive player has — something that is not observable and non-trivial to estimate. This not only reduces the complexity of the problem but also reduces the planning process to path planning.

Plan Cost Differences In Section 3, we did not provide a specific definition of the *costdiff* function. In the TTR study, we implemented a custom *costdiff* function. To generate candidate plans, we use Dijkstra's shortest path algorithm. However, players in the game can claim individual routes of a path in any order; that is, they do not need to collect routes contiguously. Thus, we define the *costdiff* function based on the number of overlapping routes between the candidate plan and an optimal plan:

$$costdiff(\pi_i, \pi_i^o) = |\pi_i \cap \pi_i^o| \quad (5)$$

Note that π_i folds in observed ontic actions, which include the routes that are already claimed. The rationale for this that a path with a higher number of claimed routes (of an optimal route) should be scored higher.

4.3 Evaluation Setup

The system requires a number of parameters. In Equation2, $\lambda = 0.8$, that is we prefer fixation length more than fixation count when computing fixation scores. The standard fixation duration threshold was set to 60ms. This is the minimum duration a player has to be fixated on a single target area for the gaze activity to be counted as one fixation. The system generates a list of inferred intentions (routes). We only look at top 10 of the inferred routes, that is, the top 10 most likely routes the player is likely to choose.

To evaluate the difference in plan recognition performance obtained with the addition of gaze data, we have a parameter to 'turn off' the gaze data. This allows us to experiment with two different models: *Model Only* system makes inferences based solely on the model-based approach with no gaze data, which serves as a baseline for our study; and the *Gaze+Model* system, which is the same but incorporates the gaze data. It then builds a candidate list of plans based on these priors and the in-game player actions.

4.4 Measures

To compare the two systems and to identify whether gaze was beneficial to intention recognition process, we used the following measures divided into three categories: (1) accuracy; (2) inference horizon; and (3) computation time. We compute the following measures using the ten most highly-ranked inferences and also using only the most likely *plan* (sequence of actions).

Accuracy. These measures are used to evaluate the overall success of inferring the ground truth and are computed by considering all inferences made up to the point in the game when the naive player completed one of the chosen tickets. We took three measures to characterise accuracy:

(1) *Recall*: This measure is the ratio of the number of correct route inferences and the total number routes claimed (ground truth) in each game. When expressed as a percentage, this shows the success of the system in inferring the ground truths, that is, the claimed routes.
(2) *Precision:* This measure is the ratio of the number of correct route inferences and the total number routes inferred, which was fixed at 10 routes per inference.
(3) *F1-Score:* This uses the previous two measures and is computed using: $F_1 = 2\times(precision\times recall)/(precision+recall)$

For these three measures, an inference was considered correct if it appeared in the top ten most highly-ranked intentions, out of a possible 79.

Inference Horizon. This measures how quickly a correct inference is made and is measured as the difference of time in seconds between when the system makes an inference, and a player makes the move corresponding to that inference and computing the averages of these times.

Computational Cost. This measures system execution time for inferring the top ten intentions.

Even though the accuracy measures do not consider where a claimed intention was ranked, we assert that this is still a useful measure, especially in the TTR game, because players will have multiple possible intentions at any one point. Our aim is not to predict the next move, but the next N moves (in this case 10 out of a possible 79). In games such as TTR, other factors such as which cards the player holds impact whether claiming a particular route is even possible. As such, we assert that accuracy is a valid measure for this evaluation.

5 RESULTS

Here we discuss the performance of *Gaze+Model* system when compared with the *Model Only* system. We tested the effects of the recognition approach on the dependent variables with a Welch's t-test. We tested the data for normality using Shapiro-Wilks tests and did not find any significant violation of normality.

5.1 Accuracy

Firstly, we analysed whether gaze data had a positive impact on inference accuracy. Table 1 shows the precision, recall and F1-score for the two systems when using top ten, five and the most likely route respectively. With a Welch's t-test, we found a significant effect of the intention recognition approach ($t(19) = 5.09$, $p < 0.01$, Cohen's d=1.14) with the Gaze+Model approach offering a higher accuracy (71%) than the Model-Only approach (47%).

Figure 3(a) shows the percentage of successful inferences of the ground truths when considering the top 10 potential routes, sorted by the *Gaze+Model* accuracy scores for presentation purposes. Clearly, gaze data had a positive impact on the accuracy of the inferences, scoring higher on accuracy in 16 of the 20 games, and scoring equally on three of the remaining four. Game 2 is the only game in which the *Model Only* approach outperformed the *Gaze+Model* approach, while in games 1, 8, and 13, the accuracy are the same for both approaches.

	Precision		Recall		F_1-Score	
N	*GM*	*MO*	*GM*	*MO*	*GM*	*MO*
N=10	0.71	0.47	0.50	0.34	0.56	0.38
N=5	0.63	0.35	0.44	0.24	0.49	0.28
N=1	0.55	0.23	0.37	0.16	0.42	0.18

Table 1: Proximal intention prediction for the two approaches. Note: GM = Gaze+Model; MO = Model Only.

While these results could be just statistical anomalies, we note that in these games, it appears that the players change their intentions rapidly and employ a number of different strategies, compared to the other games. Some collect a lot of cards before claiming routes while others claim routes as soon as they have required cards. Further, in game 2, it is clear that the players are trying to block each others intended paths, which forces a change in intention. Path blocking is an intention that was not encoded in our planner. It appears that the *Model Only* system can tackle such situations slightly better than a system that incorporates gaze data, because in the *Gaze+Model* system, the gaze data is noisy and not useful. Hence, this does raise a question: is it possible to determine when the gaze data is too noisy, and subsequently either filter out old inferences, or ignore it all together? We leave this question for future work.

5.2 Inference Horizon

We also tested whether gaze data enabled fast intention recognition. Figure 3(b) shows the inference horizon for the two approaches, with a higher inference horizon implying earlier prediction. These results show that overall the *Gaze+Model* system was able to predict the intentions quicker than the *Model-Only* system, doing so in 17 out of the 20 games. Overall the average prediction horizon of the *Gaze+Model* system was 327 seconds (149s) while the horizon for the *Model-Only* system was significantly lower at 233s (114s). With a Welch's t-test, we found a significant effect of the intention recognition approach ($t(19) = 3.31$, $p < 0.01$, Cohen's $d=0.74$) with the Gaze+Model approach recognising intentions earlier (327s) than the Model-Only approach (233s). For reasons outlined in Section 5.1, in games 1, 2, and 3 the *Model-Only* approach performs better. The post-game analysis reveals that blocking results in rapidly changing intentions and players tend to act almost immediately. We leave exploration of strategies to mitigate these for future work.

5.3 Computational Cost

The cost of the two approaches was similar. With a Welch's t-test, we found no significant effect of the intention recognition approach ($t(620) = 0.83$, $p = 0.41$, Cohen's $d=0.14$) with the planning times of the *Gaze+Model* approach (76ms (40ms)) being similar to the *Model-Only* approach (79ms (24ms)).

5.4 Summary

Our results show that incorporating gaze into intention recognition was effective in the TTR game. We compared Adding gaze meant that intentions could be inferred more accurately and earlier. However, the results also show that improvements can be made. In particular, when the players' intentions changed rapidly, the gaze data seemed to have no effect, in some case, even negative. As such, improving the model to either try to filter out older, noisy intentions or mitigate their effect, would be useful.

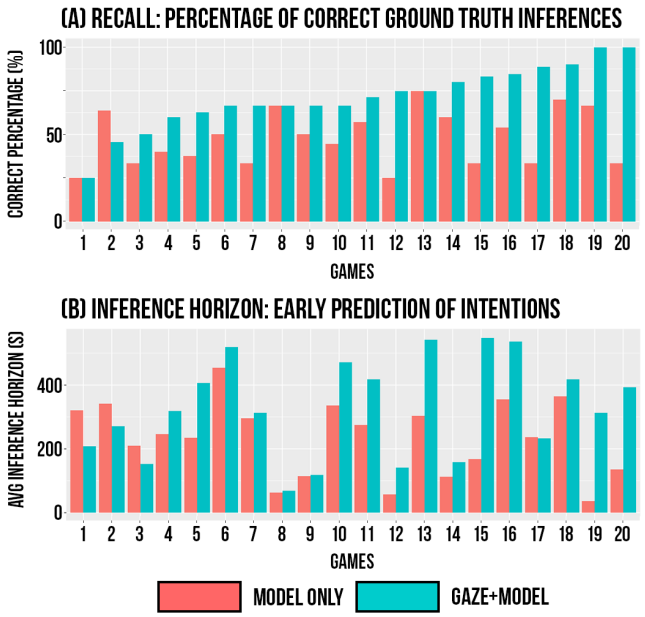

Figure 3: Comparison of accuracy and inference horizon results for the two approaches.

6 CONCLUSION

In this paper, we propose a novel model that combines gaze and model-based online intention recognition to infer intentions of humans. Gaze data is used to build probability distributions over a set of possible intentions, which are then used as priors in a model-based intention recognition algorithm.

Human-behavioural experiments demonstrated that gaze based priors significantly improved the accuracy and quickness (horizon) of the inferences when compared with classical model-based approaches. However, the model needs to be refined to cope with dynamic nature of intentions. In addition, the approach of using P-intentions to make inferences of the D-intentions was successful, at least in context of the experiments performed. These results indicate the strength of gaze-enabled model-based intention recognition.

In future work, we will extend the range of possible interaction modalities for gaining information, such as gesturing. Further, we will explore how modalities such as gaze and gesture can form part of human-agent interaction beyond intention recognition, such as collaborative planning. Another avenue of future work involves deception. The naive players in our study knew that their gaze was being monitored, but were told that their opponent could not see this data, and thus they seem to have chosen not to deceive. However, we plan to explore scenarios in which the players do employ deception, such as the 'aware' player in our study.

ACKNOWLEDGMENTS

This work was supported by two Australian Government Research Training Program Scholarship, the Microsoft Research Centre for Social NUI, and Defence Science and Technology Group CERA Grant 02.

REFERENCES

[1] Sean Andrist, Wesley Collier, Michael Gleicher, Bilge Mutlu, and David Shaffer. 2015. Look together: analyzing gaze coordination with epistemic network analysis. *Frontiers in Psychology* 6 (2015), 1016. https://doi.org/10.3389/fpsyg.2015.01016

[2] Roman Bednarik, Shahram Eivazi, and Hana Vrzakova. 2013. *A Computational Approach for Prediction of Problem-Solving Behavior Using Support Vector Machines and Eye-Tracking Data.* Springer London, London, 111–134. https://doi.org/10.1007/978-1-4471-4784-8_7

[3] Nate Blaylock and James Allen. 2004. Statistical Goal Parameter Recognition. In *Proceedings of the Fourteenth International Conference on International Conference on Automated Planning and Scheduling (ICAPS'04)*. AAAI Press, 297–304. http://dl.acm.org/citation.cfm?id=3037008.3037047

[4] Michael Bratman. 1987. *Intention, Plans, and Practical Reason.* Center for the Study of Language and Information.

[5] A. Bulling, J. A. Ward, H. Gellersen, and G. Troster. 2011. Eye Movement Analysis for Activity Recognition Using Electrooculography. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 4 (April 2011), 741–753. https://doi.org/10.1109/TPAMI.2010.86

[6] Marcus Carter, Joshua Newn, Eduardo Velloso, and Frank Vetere. 2015. Remote Gaze and Gesture Tracking on the Microsoft Kinect: Investigating the Role of Feedback. In *Proceedings of the Annual Meeting of the Australian Special Interest Group for Computer Human Interaction (OzCHI '15)*. ACM, New York, NY, USA, 167–176. https://doi.org/10.1145/2838739.2838778

[7] Philip R Cohen, C Raymond Perrault, and James F Allen. 1981. Beyond question answering. *Strategies for natural language processing* 245274 (1981).

[8] Augusto Esteves, Eduardo Velloso, Andreas Bulling, and Hans Gellersen. 2015. Orbits: Gaze Interaction for Smart Watches Using Smooth Pursuit Eye Movements. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 457–466. https://doi.org/10.1145/2807442.2807499

[9] Tom Foulsham. 2015. Eye movements and their functions in everyday tasks. *Eye* 29, 2 (2015), 196–199.

[10] Hector Geffner and Blai Bonet. 2013. A concise introduction to models and methods for automated planning. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 8, 1 (2013), 1–141.

[11] Sébastien Hillaire, Anatole Lécuyer, Gaspard Breton, and Tony Regia Corte. 2009. Gaze Behavior and Visual Attention Model when Turning in Virtual Environments. In *Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology (VRST '09)*. ACM, New York, NY, USA, 43–50. https://doi.org/10.1145/1643928.1643941

[12] Eric Horvitz and Tim Paek. 1999. A Computational Architecture for Conversation. In *Proceedings of the Seventh International Conference on User Modeling (UM '99)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 201–210. http://dl.acm.org/citation.cfm?id=317328.317354

[13] Chien-Ming Huang, Sean Andrist, Allison Sauppé, and Bilge Mutlu. 2015. Using gaze patterns to predict task intent in collaboration. *Frontiers in Psychology* 6 (2015), 1049. https://doi.org/10.3389/fpsyg.2015.01049

[14] Ryo Ishii, Ryota Ooko, Yukiko I. Nakano, and Tokoaki Nishida. 2013. *Effectiveness of Gaze-Based Engagement Estimation in Conversational Agents.* Springer London, London, 85–110. https://doi.org/10.1007/978-1-4471-4784-8_6

[15] Hiromi Kobayashi and Shiro Kohshima. 2001. Unique morphology of the human eye and its adaptive meaning: comparative studies on external morphology of the primate eye. *Journal of Human Evolution* 40, 5 (2001), 419 – 435.

[16] Kai Kunze, Yuzuko Utsumi, Yuki Shiga, Koichi Kise, and Andreas Bulling. 2013. I Know What You Are Reading: Recognition of Document Types Using Mobile Eye Tracking. In *Proceedings of the 2013 International Symposium on Wearable Computers (ISWC '13)*. ACM, New York, 113–116.

[17] Peta Masters and Sebastian Sardina. 2017. Cost-Based Goal Recognition for Path-Planning. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems (AAMAS '17)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 750–758. http://dl.acm.org/citation.cfm?id=3091125.3091232

[18] J. Munoz, G. N. Yannakakis, F. Mulvey, D. W. Hansen, G. Gutierrez, and A. Sanchis. 2011. Towards gaze-controlled platform games. In *2011 IEEE Conference on Computational Intelligence and Games (CIG'11)*. 47–54. https://doi.org/10.1109/CIG.2011.6031988

[19] Joshua Newn, Fraser Allison, Eduardo Velloso, and Frank Vetere. 2018. Looks Can Be Deceiving: Using Gaze Visualisation to Predict and Mislead Opponents in Strategic Gameplay. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA. https://doi.org/10.1145/3173574.3173835

[20] Joshua Newn, Eduardo Velloso, Fraser Allison, Yomna Abdelrahman, and Frank Vetere. 2017. Evaluating Real-Time Gaze Representations to Infer Intentions in Competitive Turn-Based Strategy Games. In *Proceedings of the Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '17)*. ACM, New York, NY, USA, 541–552. https://doi.org/10.1145/3116595.3116624

[21] Joshua Newn, Eduardo Velloso, Marcus Carter, and Frank Vetere. 2016. Exploring the Effects of Gaze Awareness on Multiplayer Gameplay. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play Companion Extended Abstracts (CHI PLAY Companion '16)*. ACM, New York, NY, USA, 239–244. https://doi.org/10.1145/2968120.2987740

[22] Elisabeth Pacherie. 2008. The phenomenology of action: A conceptual framework. *Cognition* 107, 1 (2008), 179 – 217. https://doi.org/10.1016/j.cognition.2007.09.003

[23] Ramon Pereira, Nir Oren, and Felipe Meneguzzi. 2017. Landmark-Based Heuristics for Goal Recognition. In *AAAI*. https://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14666

[24] Miquel Ramírez and Hector Geffner. 2010. Probabilistic Plan Recognition Using Off-the-shelf Classical Planners. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI'10)*. AAAI Press, 1121–1126. http://dl.acm.org/citation.cfm?id=2898607.2898786

[25] Gita Sukthankar, Christopher Geib, Hung Hai Bui, David Pynadath, and Robert P. Goldman. 2014. *Plan, Activity, and Intent Recognition: Theory and Practice* (1st ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[26] Milind Tambe and Paul S. Rosenbloom. 1995. RESC: An Approach for Real-time, Dynamic Agent Tracking. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1 (IJCAI'95)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 103–110. http://dl.acm.org/citation.cfm?id=1625855.1625869

[27] Michael Tomasello, Brian Hare, Hagen Lehmann, and Josep Call. 2007. Reliance on head versus eyes in the gaze following of great apes and human infants: the cooperative eye hypothesis. *Journal of Human Evolution* 52, 3 (2007), 314 – 320.

[28] Eduardo Velloso and Marcus Carter. 2016. The Emergence of EyePlay: A Survey of Eye Interaction in Games. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '16)*. ACM, New York, NY, USA, 171–185.

[29] Mor Vered, Gal A. Kaminka, and Sivan Biham. 2016. Online Goal Recognition through Mirroring: Humans and Agents. In *Proceedings of the Fourth Annual Conference on Advances in Cognitive Systems.* A slightly modified version appears in Proceedings of the IJCAI 2016 workshop on Human-Agent Interaction Design and Models (HAIDM).

[30] Stefanie Wetzel, Katharina Spiel, and Sven Bertel. 2014. Dynamically Adapting an AI Game Engine Based on Players' Eye Movements and Strategies. In *Proceedings of the 2014 ACM SIGCHI Symposium on Engineering Interactive Computing Systems.* ACM, New York, NY, USA, 3–12.