Eye-Tracking for User Modeling in Exploratory Learning Environments: an Empirical Evaluation

Cristina Conati [1,2] and Christina Merten[1]
[1]Department of Computer Science, University of British Columbia
2366 Main Mall, Vancouver, BC, V6Z2T4, Canada
Tel: 01-604-8224632. FAX: 01-604-822-5485
conati@cs.ubc.ca

[2] Department of Information and Communication Technology, University of Trento
Via Sommarive 14, 3805, Povo, Trento, Italy

ABSTRACT

In this paper, we describe research on using eye-tracking data for on-line assessment of user meta-cognitive behavior during interaction with an environment for exploration-based learning. This work contributes to user modeling and intelligent interfaces research by extending existing research on eye-tracking in HCI to on-line capturing of high-level user mental states for real-time interaction tailoring. We first describe the empirical work we did to understand the user meta-cognitive behaviors to be modeled. We then illustrate the probabilistic user model we designed to capture these behaviors with the help of on-line information on user attention patterns derived from eye-tracking data. Next, we describe the evaluation of this model, showing that gaze-tracking data can significantly improve model performance compared to lower level, time-based evidence. Finally, we discuss work we have done on using pupil-dilation information, also gathered through eye-tracking data, to further improve model accuracy.

KEYWORDS: User Modeling, Eye Tracking, Intelligent Learning Environments, Exploration-based Learning, Self-explanation

CORRESPODING AUTHOR: Cristina Conati. Phone Number until January 6th, 2007: +39-339-449-1400 (after January 6, see contact info above)

INTRODUCTION

One of the functionalities that an Intelligent User Interface may include is providing tailored support to help users perform complex tasks. Providing this functionality involves building a model of user traits relevant to adequately tailoring the interaction, i.e., a *user model*. Depending on the nature of the task and the extent of the support, the relevant user traits may include simple performance measures (such as frequencies of interface actions), domain-dependent cognitive traits (such as knowledge and goals), meta-cognitive processes that cut across tasks and domains, and affective states. Arguably, the higher the level of the traits to be captured, the more difficult it is to assess them unobtrusively from simple interaction events. This problem has generated a stream of research on using innovative sensing devices to enrich the information available to a user model.

Our work contributes to this research stream by presenting results on using real time eye-tracking information to inform a user model designed to assess student meta-cognitive behavior during interaction with an *Exploratory Learning Environment* (ELE). ELEs are computer-based educational tools designed to stimulate learning through free exploration of the target domain, instead of through the more structured activities supported by traditional Intelligent Tutoring Systems [29].

The meta-cognitive behaviors covered by the user model we describe include the capability to learn effectively from free exploration [11, 30] and the capability to *self-explain* instructional material, i.e., to clarify and elaborate the given information in light of the underlying domain theory (e.g., [7] and [25]). Both of these meta-cognitive skills have been shown to improve the quality of student learning with ELE, but it has also been shown that many students lack them [7, 25].

As a consequence, there have been several efforts to support the acquisition of these skills in ELE. However, few of these efforts tried to generate support tailored to student meta-cognitive needs. In relation to effective exploration, most work focused on providing interface tools that stimulate the right exploratory behaviors [23, 29]. In relation to self-explanation, research either focused on generating untailored prompts [1], or relied on simple tailoring strategies such as prompting for self-explanation after every new action or after every incorrect action [21]. One of the reasons for the lack of sophisticated tailoring is the difficulty of assessing user meta-cognitive behaviors. Conati and Vanlehn [10] proposed a system that models user self-explanation behavior using interface artifacts that allow the system to obtain relevant information on user attention. However, it is not always possible to devise interface artifacts that do not interfere with the nature of the interaction. For this reason, we are exploring the use of eye-tracking data to provide information on user meta-cognition. In particular, in this paper we describe our inclusion of eye-tracking information to track student self-explanation and exploration behavior in ACE (Adaptive Coach for Exploration), an ELE that supports student exploration of mathematical functions.

In an ELE, an essential component of effective exploration involves trying out a variety of domain-exploration actions. However, simply trying out actions is not sufficient for learning. It is also important to attend to and reason about the actions' outcome, i.e, to perform self-explanation. Intuitively, self-explanation may be more likely if the student actually attends to the parts of the interface showing the effects of a specific exploratory action. Previous studies [22] showed that a user's gaze can be a good indicator of which parts of an interface hold the user's attention. Thus, it is reasonable to assume that the

presence or absence of certain gaze patterns (such as *gaze shifts* from the screen region where an action is executed to the region showing its effects) may be used to assess self-explanation behavior. The first objective of this work is to test this assumption. The second objective is to show that accurate assessment of self-explanation can improve assessment of student exploratory behavior. To meet these objectives, we set two subgoals: (i) creating a student model that uses specific gaze patterns as evidence of implicit self-explanation; (ii) evaluating the performance of the model as a predictor of implicit self-explanation and sufficient exploration, particularly in comparison with previous model versions that do not use gaze-tracking data. In this paper, we describe how we achieved both goals. For goal (i), we first illustrate a user study that tests assumptions on which gaze patterns indicate implicit self-explanation and compare these patterns as predictors against a simpler measure based on action latency. We then describe the user model we built based on the results of this study.

For goal (ii), we show that including gaze-tracking information significantly improves the model assessment of student self-explanation, compared to previous versions. The evaluation also shows that more accurate assessment of student self-explanation significantly improves the assessment of student learning through exploration. This is a significant contribution to the research on eye-tracking in HCI, which has mostly involved the use of eye-tracking for either interface evaluation/manipulation or on-line assessment of lower-level mental states.

We also explored whether pupil dilation information may further improve model accuracy. Previous research found a positive correlation between a person's pupil dilation and cognitive load in a wide variety of tasks [4]. Self-explanation requires cognitive effort to make sense of studied instructional material, thus students who self-explain may incur a higher cognitive load than students who do not. Since the eye-tracker we used in this research provides data on pupil size, we investigated whether this measure could be an additional predictor of self-explanation for our student model. Our findings contribute to the increasing body of work that has been recently devoted to evaluate the performance of pupil dilation as a source of information for interface adaptation.

In the rest of the paper, we first discuss related work. Next, we introduce the ACE learning environment, the ELE we used in this project. Then we present previous versions of the ACE student model and their limitations, and we follow with the illustration of the new model. Next, we describe a a study to evaluate the new model. Finally, we present our investigation on whether pupil dilation information, also derived from eye-tracking data, may contribute to model accuracy. We conclude with a discussion of future work

RELATED WORK

Research on self-explanation and exploration

Several studies in Cognitive Science have shown the effectiveness of *self-explanation* as a learning skill in a variety of instructional tasks, including studying worked-out example solutions (e.g., [8] [25]), reading instructional text ([8]) and solving problems (e.g., [1]). Because there is evidence that this learning skill can be taught (e.g., REF), several computer-based tutors have been devised to provide explicit support for self-explanation. However, all of these tutors focus on coaching self-explanation during fairly *structured* interactions targeting problem-solving skills (e.g., [1], [10] and [21]). For instance, the SE-Coach [10] is designed to model and trigger students' self-explanations as they study examples of worked-out solutions for physics problems. The Geometry Explanation Tutor [1] and Normit-SE [21] support self-explanations of problem-solving steps, in geometry theorem proving and data normalization respectively. In this paper, we describe our extension of support for self-explanation to the *less structured* pedagogical interactions supported by exploratory learning environments.

Exploratory learning environments place less emphasis on supporting learning through structured, explicit instruction and more on allowing the learner to freely explore the available instructional material (e.g., [11, 30], [29]). In theory, this type of active learning should enable students to acquire a deeper, more structured understanding of concepts in the domain. In practice, empirical evaluations have shown that open learning environments are not always effective for all students (e.g., [11, 30], [11, 30]). The degree of learning from such environments depends on a number of student-specific features, such as activity level and whether or not the student already possesses the meta-cognitive skills necessary to learn from exploration. These results highlight the importance of providing support to exploration-based learning that is tailored to the needs of individual students.

Eye-tracking research

Retrospective analysis of eye movement

In HCI, retrospective analysis of eye movement data has been studied to evaluate usability issues and understand human performance. For instance, Schiessl *et al*. [27] used an eye-tracker to investigate gender differences in attention behavior for textual *vs.* pictorial stimuli on websites. An interesting outcome was that, when the participants were asked where in the interface they thought they looked, their perceptions often differed from reality, showing that accurate attention patterns could only be found with an eye-tracker. In [13], offline processing of eye-tracking data was used to improve the efficient generation of non-photorealistic images. Users' eye fixations were analyzed to determine which parts of given pictures users found to be most meaningful, and the findings were used to design algorithms that draw the most "important" parts of the picture first.

The research described in this paper differs from the efforts above because, although it includes the use of retrospective analysis of eye movements to design and test a student model, it also uses eye-tracking data *on-line* to model student learning.

On-line use of eye-tracking in interface operation

There has also been fairly extensive research in using eye gaze as an alternative form of input to allow a user to explicitly operate an interface. In [19], Jakob explores issues surrounding the real-time processing of eye data such as efficient noise reduction and the organization of gaze information into tokens from which relevant data may be extracted. He then discusses the potential of eye-tracking as a tool in several forms of

interface manipulation, including object selection/movement, scrolling text, and navigating menus. Salvucci and Anderson [26] applied these ideas to design IGO (Intelligent Gaze-added Operating-system), a system that allows users to use their eyes to perform interface operations such as opening, closing and dragging windows. Majaranta *et al*. [20] devised a system that allowed users to type with their eyes via an eye-tracker, given a picture of a keyboard for users to look and an algorithm to interpret small fixations as key presses. In [16], Hornof *et al*. describe EyeDraw, a system to enable children with severe motor impairments to draw pictures by just moving their eyes. Unlike the above systems, the work discussed in this paper uses real-time processing of a user's gaze to interpret user non-explicit meta-cognitive behaviours for on-line interaction adaptation.

Eye-tracking for on-line interaction adaptation

A parallel research stream has used eye-tracking data on-line for real-time interaction adaptation. Some of this work uses gaze tracking to assess user *task performance*. For example, in [31], Sibert *et al.* describe the use of gaze tracking to assess reading performance in the Reading Assistant, a system for automated reading remediation that provides visual and auditory cues if user gaze patterns indicate difficulties in reading a word. In [17]. Iqbal and Bailey use gaze-tracking to determine which type of task the user is performing (e.g., reading email *vs.* reading a web page), with the goal of devising an attention manager that balances the user's need for minimal disruption with an application's need to deliver necessary information.

There has also been research on using gaze information for real-time adaptation to user *mental states* such as interest or problem-solving strategies. In [32], Starker and Bolt describe a system that uses an eye-tracker to determine which part of a graphical interface a user is interested in, and then provides more information about this area via visual zooming or synthesized speech. In [24], Qu and Johnson use eye-tracking for interaction adaptation within the Virtual Factory Teaching System (VFTS), an computer tutor for teaching engineering skills. Eye-tracking is used to discern the time the user spends reading something from the time the user spends thinking before taking action, with the goal of assessing and adapting to the motivational states of student effort and confusion. Gluck and Anderson [15] studied the use of eye-tracking to assess student problem-solving behaviors within the PAT Algebra I tutor, including attention shifts, disambiguation of problem statements and errors, processing of error messages and other information critical to problem solving.

Our work extends this body of research by exploring if and how eye-tracking can help assess mental states related to the meta-cognitive, domain-independent skill of self-explanation.

Using pupil dilation in adaptive interfaces

Recently, there has been increasing interest in exploring the potential of pupil dilation as a source of information for an adaptive system, mostly because of the link that has been found between pupil dilation and cognitive load [3]. However, so far the existing research on this topic has yielded controversial results. Schultheis and Jameson [28] analyzed pupil sizes of users reading texts of varying difficulty within an adaptive hypermedia system. They found that the difference in text difficulty – and thus cognitive load – was not reflected in pupil diameter changes. Iqbal *et al.* [18] examined the sensitivity of pupil size to cognitive load as users performed different tasks, including file manipulation and the reading of text. While, as in [28], pupil size failed to be an accurate indication of cognitive load during the reading tasks, it was found to be sensitive to task difficulty during certain subtasks of the file management task.

The investigations described in this paper contributes to the above body of work by providing initial indications that pupil dilation is not a good predictor of self-explanation during exploration with an ELE.

THE ACE LEARNING ENVIRONMENT

ACE is a learning environment that supports the exploration of mathematical via a set of activities divided into units and exercises. Units are collections of exercises whose material is presented with a common theme and mode of interaction. Exercises within units differ in function type and equation.

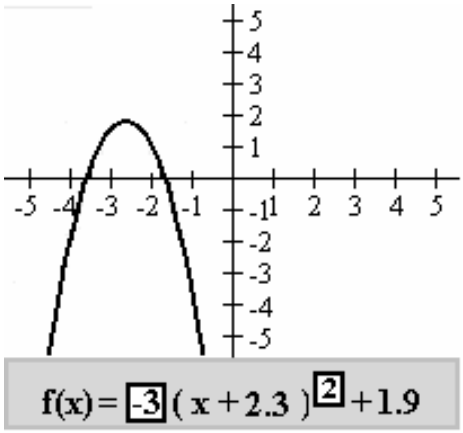

Figure 1: ACE's Plot Unit

Figure 1 shows the main interaction window for the Plot Unit. We will focus on this unit throughout the paper because it is the unit most relevant to the research presented in later sections. In the Plot Unit, a learner can explore the relationship between a function's graph and equation by moving the graph in the Cartesian plane and observing how that affects the equation (displayed below the graph area). The student can also change the equation parameters and see how these changes affect the graph.

To support the exploration process, ACE includes a coaching component that provides tailored hints when ACE's student model detects that students are having difficulties exploring effectively. For more details on ACE's interface and coaching component, see [5]. In the next section, we describe two preliminary versions of ACE's student model, which we will use for performance comparison against the new model with eye-tracker data in the evaluation section.

PREVIOUS VERSIONS OF ACE'S STUDENT MODEL

Version with no self explanation

ACE's student model uses Dynamic Bayesian Networks (DBN) [12] to manage the uncertainty in assessing students' exploratory behavior. The main cause of this uncertainty is that the reasoning processes that influence the effectiveness of student exploration are not easily observable unless students are required to make them explicit. However, forcing students to articulate this reasoning would likely be intrusive and clash with the unrestricted nature of this type of learning.

We chose DBNs for the ACE student model because they are a well-established formalism for reasoning under uncertainty in domains where there are *dynamic* random variables, i.e., variables with values that can change over time (e.g., student knowledge during interaction with ACE). A DBN is a graph where nodes represent stochastic variables of interest and arcs capture the direct probabilistic relationships between these variables, including temporal dependencies between the evolving values of dynamic variables. Each node has an associated probability distribution representing the conditional probability of each of its possible values, given the values of its parent nodes. As evidence on one or more network variables becomes available, *ad hoc* algorithms update the posterior probabilities of all the other variables, given the observed values

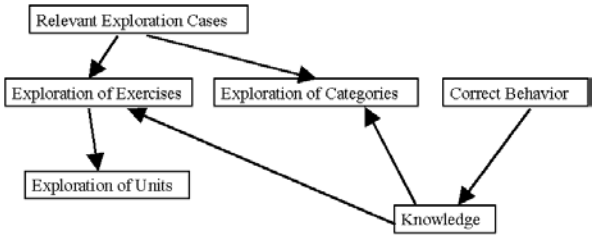

Figure 2. High-level Structure of ACE's Student Model

The first version of ACE's student model was derived from an iterative design process that yielded a better understanding of what defines effective exploration [5]. Figure 2 shows a high-level description of this model, which comprises several types of nodes to assess exploratory behaviour at different levels of granularity. These nodes include: Relevant *Exploration Cases*, representing exploration of individual exploration cases in an exercise (e.g., changing the slope of a line to 3, a positive number, in the Plot Unit); *Exploration of Exercises* and *Exploration of Units,* representing adequate exploration for the various ACE exercises and units, respectively; and *Exploration of Categories,* representing the exploration of groups of relevant exploration cases that appear across multiple exercises (e.g., all of the exploration cases involving a positive slope in the Plot Unit). The links among the different types of exploration nodes represent how they interact to define effective exploration. Exploration nodes have binary values representing the probability that the learner has sufficiently explored the associated items.

ACE's student model also includes binary nodes representing the probability that the learner understands the relevant pieces of knowledge (summarized by the node *Knowledge* in Figure 2). The links between knowledge and exploration nodes represent the fact that the degree of exploration needed to understand a concept depends on how much knowledge a learner already has. Knowledge nodes are updated only through actions for which there is a clear definition of correctness. These nodes are never updated within the Plot Unit, since it consists of purely exploratory activities.

Initial studies on ACE generated encouraging evidence that the system based on the model in Figure 2 could help students learn better from exploration [5]. However, these studies also showed that sometimes the ACE student model – labeled *Action_Based model* from now on – overestimated students' exploratory behaviour, because it considered interface actions to be sufficient evidence of good exploration, without taking into account whether a student was *self-explaining* the outcome of these actions. For instance, a student who quickly moves a function graph around the screen in the Plot Unit – but never reflects on how these movements change the function equation – performs many exploratory actions but can hardly learn from them because she is not reflecting on (self-explaining) their outcomes. We observed this behavior in several study participants.

Extending ACE to Track and Support Self-Explanation

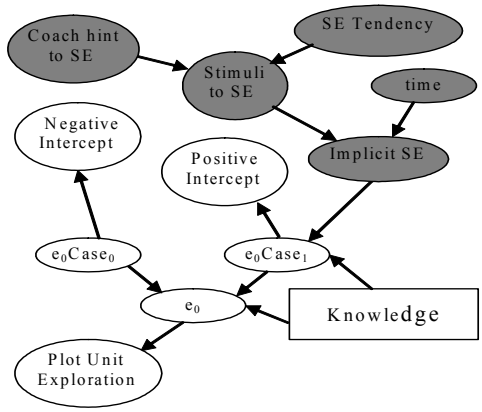

Figure 3. Original ACE student model with self-explanation

To address the model limitation described above, we started extending ACE's interface and student model to track and support self-explanation [6]. The original version of ACE only generated hints indicating that a student should further explore some elements of a given exercise. Augmenting ACE with the capability to track self-explanation allows ACE not only to detect when a student's exploration is sub-optimal, but also to understand if the cause is a lack of self-explanation and generate tailored hints to correct this behavior.

There are two types of self-explanation that ACE needs to detect: (i) *explicit* self-explanation, i.e., self-explanation that the student generates using menu-based tools available in the interface; and (ii) *implicit* self-explanation that students generate in their head. The latter is the most difficult to detect, due to the

lack of hard evidence of its occurrence. Implicit self-explanation is the focus of the extensions to the student model we describe in the next sections. The first version of the ACE student model with an assessment of self-explanation [6] used only time spent on each exploratory action as evidence of implicit self-explanation. We will refer to this model as Time_Based model from now on. Figure 3 shows a time slice1 in this model, corresponding to an implicit self-explanation action (similar slices capture the occurrence of explicit self-explanation).

Nodes representing the assessment of self-explanation are shaded grey. In this figure, the learner is currently exploring exercise 0 (node e0) in the Plot Unit, for which two relevant exploration cases (e0Case0 and e0Case1 in Figure 3) are shown. Each exploration case influences one or more exploration categories (positive intercepts and negative intercepts in the figure). Here the learner performs an action corresponding to e0case1. In this new version of the model, the probability that a learner's action implies effective exploration of a given case depends on both the probability that the student self-explained the action and the probability that she knows the corresponding concept, as assessed by the set of knowledge nodes in the model (summarized in Figure 3 by the node Knowledge). Factors influencing the probability that implicit self-explanation occurs include the time spent exploring the case and the stimuli that the learner has to self-explain. Low time is always taken as negative evidence for implicit explanation. The probability of self-explanation with longer time spent on an action depends on whether there is a stimulus to self-explain, i.e., on the learner's general tendency to self-explain and on whether the system generated an explicit hint to self-explain.

Time, however, can be an ambiguous predictor for self-explanation. First, it is hard to define for different learners what is insufficient time for self-explanation. Furthermore, a student may be completely distracted during a long interval between exploration cases. Thus, we chose to explore an additional source of evidence of self-explanation behavior, i.e., the student's attention patterns during the exploration of a given case.

ADDING EYE-TRACKING TO ACE – PRELIMINARY INVESTIGATION

The intuition for using an eye-tracker to assess self-explanation is that self-explanation is more likely to have occurred if the student actually attends to the interface's regions showing the effects of a specific exploratory action. As an example, if a student has modified the function equation, a gaze shift pattern suggestive of self-explanation would start from the equation region and then hover around the graph region above.

However, monitoring user gaze with an eye-tracker can be expensive and laborious. Thus, we ran a study to compare the effectiveness of gaze tracking as predictor of self-explanation against a simpler time-based predictor. Although the study was discussed in [8] we report it here, expanding on the experimental design and data analysis, because it lays the groundwork for the new model and evaluation methodology described later.

Study Design

The 19 participants recruited for the study were university students who had not yet taken any college level math, but had had high school calculus. We set this requirement to have subjects with limited knowledge of mathematical functions, so that it would be meaningful for them to explore this topic through ACE. We needed subjects with some initial understanding of the subject, however, because ACE does not offer any background instruction and total novices wouldn't know where to start.

The study consisted of individual sessions lasting approximately 80 minutes. In each session, the student first completed a pre-test to determine his/her initial knowledge of mathematical functions. This was followed by interaction with ACE2, while a student's eye movements were recorded by an head-mounted eye-tracker. This particular eye-tracker was used because it was readily available through the Psychology Department at the University of British Columbia. However, the same data could be obtained through a completely non-intrusive remote eye-tracker, consisting of a small camera that sits on top of the monitor or on some other flat surface (e.g., IView X Red from SensoMotoric Instruments, USA).

Before starting the interaction with ACE, participants underwent an eye-tracker calibration phase. Next, we used a standard script to instruct participants to "think aloud", i.e., to verbalize what they thought during interaction, even if it seemed unimportant. Finally, participants went through each of the ACE units at their own pace. After exiting ACE, each participant completed a post-test very similar to the pre-test, the only differences being the constants used in the functions and the ordering of different questions.

In addition to the paper pre-test and post-test, each session yielded a log file from the ACE system, which included each exploratory action and the time when it was taken. The eye-tracker also generated a log file containing the coordinates and duration of each fixation. In addition, we collected video and audio recordings of the interaction, showing the ACE screen and allowing for later analysis of the user's speech.

Data Analysis

Setting the "gold standard"

In order to assess the performance of latency and gaze patterns as predictors of implicit self-explanation, it was necessary to first classify students' self-explanation behavior. In particular, we needed to isolate exploratory actions accompanied by implicit self-explanation (termed *positive* self-explanation cases from now on) and those that were not (termed *negative* self-explanation cases), so that these classifications could be tested for correlations with those predicted by time and gaze shifts. Note that here "negative self-explanation" indicates situations in which the students did not self-explain, not

[1] In a DBN, a time slice represents the model's variables at a particular point in time. The temporal evolution of dynamic variables is represented by a sequence of time slices connected by links that encode the temporal dependencies between slices.

[2] In this study, we used the original version of ACE, which does not include tools to support explicit student self-explanation.

situations in which students self-explained incorrectly, consistent with the original definition of self-explanation [8].

We used the audio recordings of each interaction for detecting the presence or absence of implicit self-explanation. As described earlier, these recordings consisted of explicit verbalization by the subjects as to their thoughts. Similar to other studies on self-explanation [3, 21], using subjects' verbalizations is acceptable since the participants were instructed to share all of their thoughts and were not told anything about the data analysis process or the actual purpose of the study. Thus, the episodes we related to presence or absence of self-explanation in the data can accurately be described as "internal" since they reflect the subjects' thoughts, which are unknown to the ACE system. Further, with existing technology, this is as close as we could come to reading the students' thoughts in our search for evidence of implicit self-explanation or lack thereof.

To maximize objectivity in the analysis of the audio data, two observers (the second author and another graduate research assistant) independently analyzed the audio data and then created links between the verbal episodes and the corresponding exploration cases in the log files. This turned out to be a fairly laborious process, because of two main factors.

First, we needed to devise a detailed coding scheme in order to objectively convert fragments of audio data into isolated episodes of positive or negative self-explanation. While coding schemes exist for self-explanation study during problem solving (see, for instance, [8]), ours was the first attempt to evaluate self-explanation during independent exploration. This problem was addressed by having the two observers independently label a subset of the audio data, then compare their classifications, possibly reconcile them and devise the coding scheme based on this discussion.

Table 1: Coding scheme for self-explanation episodes

Evidence of positive self-explanation	Evidence of negative self-explanation
Verbalized conclusions about domain-specific principles related to the exploration process (regardless of correctness) Prediction of an action just before it occurred	Simple narration of the interaction Isolated statements of confusion Expressions of inattentiveness

In the coding scheme, student utterances were classified as self-explanation if they expressed a conclusion about a domain-specific principle related to the exploration process (e.g., "*when I increase the coefficient here, the line gets steeper*") regardless of correctness, or if they predicted the result of an action just before it occurred (e.g., "*putting a negative sign here will turn the curve upside-down*"). Simply narrating the outcome of each action once it happened (e.g., "*this number just changed to a 3*"), obvious statements of inattentiveness (e.g., "*I'm just playing*") or isolated statements of confusion (e.g., "*I don't understand what's happening*") were not considered self-explanation. However, tentative explanations followed by expressions of confusion were coded as self-explanation. This classification scheme is summarized in Table 1. It should also be noted that whenever an exploratory action was followed by evidence of both positive and negative self-explanation, the action was considered self-explained. The coded data for two episodes appears below.

CODED DATA FOR TWO VERBALIZATION EPISODES

```
<ACE_TIME: 16:59:23> <VID_TIME:02:38:34>
(a)  <ACE_TIME: 17:07:08> <VID_TIME:> <ACTION: Moved constant function> <SE_TYPE:N> <SE_DESCRIPTION: "I'm not sure what's going on"> <LOG_PTR: 1057>
(b)  <ACE_TIME: 17:07:27> <VID_TIME:> <ACTION: Moved linear function> <SE_TYPE:Y> <SE_DESCRIPTION: "moving the line changes the y intercept in the equation"> <LOG_PTR: 1127>
```

Here, the various tags describe each episode, as follows. *ACE_TIME* gives the system time when the action occurs; *VID_TIME* gives the time as kept by the video-recorder; *ACTION* describes the exploratory action occurred; *SE_TYPE* gives the aforementioned experimenters classification of self-explanation episodes. Here a Y (yes), N (no) or ? (inconclusive) always appears; *SE_DESCRIPTION* gives the student's statement used to define SE_TYPE; *LOG_PTR* gives the action's line number in the ACE log file for quick reference

For each coded episode, the experimenter records the relevant utterance in the SE_DESCRIPTION tag, as shown in the two coded episodes above. The VID_TIME tag gives the video-recorder time of when the speech occurred, and the SE_TYPE tag gives the corresponding observers' classification. A post-processing program helps find the ACE action associated with a given utterance by using the synchronization line appearing at the start of each coded file, which gives the time when ACE was started in both forms. From this line, the program derives the ACE_TIME for each coded utterance from its corresponding VID_TIME tag, and then retrieves the co-occurring action from the ACE log file, filling the ACTION, and LOG_PTR tags accordingly.

The second factor contributing to the complexity of data analysis was that knowing the time of occurrence was not always sufficient to map utterances with actions. The observers at first assumed that subjects' utterances always pertained to whatever exploratory action they had just taken. However, subsequent analysis of the video data showed that this was not always the case, particularly for users who showed great reluctance to think aloud. These learners had to be repeatedly prompted by the observers to speak, so some of the conclusions they shared weren't reached when they spoke, but related to self-explanation that occurred a few minutes earlier. The observers solved this problem by studying every coded episode and using its content to match it to its corresponding action. For example, if a user made a comment about even exponents, it was matched with an exploratory action which involved even exponents, even if this action occurred slightly earlier. Thirteen coded episodes were discarded because the match was ambiguous.

Each observer individually applied the above coding criteria to code the audio data, and then their results were compared. The intercoder reliability was 93% , which suggests a high level of objectivity in the classification scheme. Only episodes on

which the coders fully agreed were used in the rest of the analysis.

While all the factors mentioned above resulted in the elimination of data points, the factor that had the greatest impact on the amount of data that could be obtained from the study was students' willingness to verbalize their thoughts. A number of students were incapable or unwilling to think aloud, even if they were periodically reminded to do so. Without such verbalization, the coders could not tell whether a student had self-explained or not. Thus, of the 567 exploration cases recorded in the log files for all students, only 149 could be classified in terms of associated self-explanation and used to explore the correspondence between self-explanation, gaze information and time, as described next[3].

Time and gaze shifts as predictors of self-explanation

To analyze the relationship between time per exploration case and self-explanation, we first checked whether there was any difference between average time spent on exploration cases with self-explanation (24.7 seconds) and without (11.6 seconds). The difference is statistically significant at the 0.05 level (two-tailed t-test), suggesting that time per exploration case is a fairly reliable indicator of self-explanation. We then used ROC curve analysis to determine the optimal threshold to indicate sufficient time for self-explanation, which we determined to be 16 seconds (see [9] for more details). The reader should recall that both positive and negative self-explanations are verbalized, so higher time for positive self-explanation is not an artifact of verbalization.

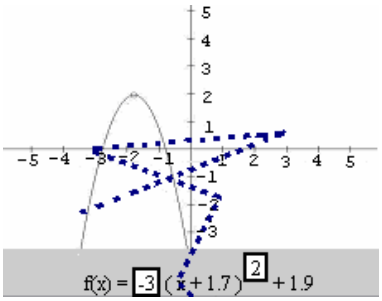

Figure 4: Sample gaze shift

Raw eye-tracker data was parsed by a pattern detection algorithm developed to detect gaze patterns we hypothesized to be associated with self-explanation in the plot unit[4]. These patterns consist of switches of attention ("gaze shifts") between the graph panel and the equation area. We considered as gaze shifts both direct switches of attention between the plot and equation regions, as well as switches where the gaze moves to non-salient regions in between (indirect gaze shifts). We did so because we believe that both types of shifts can indicate student attention to the relationship between changes in the function equation and in its plot[5]. A sample (direct) gaze shift appears in Figure 4. Here a student's eye gaze (shown as the dotted line) starts in some untracked area below the screen, moves to the equation region and then hovers around the graph region above. The data-parsing algorithm uses fixation coordinates from the eye-tracker and matches them to appropriate ACE interface regions. Next, it searches the data for the pattern of looking at one region and then another, i.e., having a gaze shift. In post-processing mode, when a gaze shift is found, a tag is placed in the ACE log file to synchronize the switch with the appropriate exploration case[6].

An excerpt of a synchronized log file appears in Figure 5. Here, the user begins by typing a new value for the slope into the equation (line 1 in the figure).

```
1 <EXPLORE><TEXT BOX>*17:11:22*new_slope
8
2 <INDIRECT FIX
CHANGE>*17:11:23*Previous Region:
                Equation
 3 <DIRECT FIX CHANGE>*17:11:24*Previous
              Region: Graph
 4 <DIRECT FIX CHANGE>*17:11:24*Previous
            Region: Equation
    5 <EXPLORE>*17:11:26*Moved power
         function to: new Coeff:
  8.0 new yInt: 2.02 xInts:
1.3533333333333333
6 <EXPLORE>*17:12:19*Moved power
function to: newCoeff:
  8.0 new yInt: 2.02 xInts: -
0.819999992847464
7 <DIRECT FIX CHANGE>*17:12:24*Previous
Region: Graph
8 <DIRECT FIX CHANGE>*17:12:25*Previous Region: Equation
```

Figure 5: Excerpt of ACE log file with added gaze shift tags

Then several gaze shifts occur (lines 2-4). Log entries for gaze shifts only report the region where the shift originated, since the region where the gaze shifted to is implicit in the definition of gaze shift. So, for instance, the log entry in line 2 represents an *indirect gaze shift* that starts from the equation area, moves to an (unreported) irrelevant region (e.g., ACE help menu on the screen or the keyboard on the table) and then ends in the graph region. The next two entries represent two consecutive *direct gaze shifts* from the graph to the equation area and back. Next the user moves the curve without looking down at the function region (line 5). Finally, the user moves the curve again and then shifts her gaze down to the function region and back (lines 6-8).

[3] Each of the 36 participants generated at least some relevant verbalizations, thus no student was eliminated by this process.

[4] This algorithm was devised by Dave Ternes, an undergraduate research assistant in the computer science department at UBC.

[5] Although we also believe that the distinction between direct and indirect gaze shifts may be utilized to make finer-grain inferences on student self-explanation, we felt we did not have enough data to explore the differences in this work. However, we kept the distinction in the log files for future work.

[6] In on-line processing mode, the detection of a gaze shift or lack thereof is passed as evidence to the ACE student model, as we will describe in a later section.

After the synchronized log file has been generated, a program merges it with the coded data from the observers to create a file containing all data for one user in a concise form.

Results and Discussion

Table 2: Classification Accuracy of different predictors

	Eye-tracker	Time	Eye-tracker + Time
True Positive Rate (*sensitivity*)	61.6%	71.7%	85.8%
True Negative Rate (*specificity*)	76.0%	68.0%	62.0%
Combined Accuracy	68.8%	69.85%	73.9%

Table 2 shows different measures of self-explanation classification accuracy if the predictor used is: (i) the eye-tracker to detect gaze shifts; (ii) time per exploration case, where the occurrence of self-explanation is predicted if time is greater than the 16 seconds threshold; (iii) a combination of the two, where the occurrence of self-explanation is predicted if there is a gaze shift *or* time is greater than the threshold[7]. Accuracy is reported in terms of true positive rate (i.e., percentage of positive self-explanation cases correctly classified as such, or sensitivity of the predictor), true negative rate (i.e., percentage of negative self-explanation cases correctly classified as such, or the specificity of the predictor) and the average of these two accuracies. As the table shows, time alone has a higher *sensitivity* than gaze shift, i.e., the episodes involving self-explanation were more likely to take over 16 seconds than to include a gaze shift[8]. However, the eye-tracker alone has comparably higher *specificity*, i.e., the cases without self-explanation were more likely to involve the absence of a gaze shift than shorter time per exploration case. The two predictors have comparable combined accuracy. Alternate analysis was performed to check if multiple gaze shifts would serve as a good predictor, with or without time. When two gaze shifts were required to indicate self-explanation, the specificity of the eye-tracker alone dropped to 51.7%, and the sensitivity only rose to 79.5%, resulting in a combined accuracy of 60.6%. Adding time raised the combined accuracy to 67.9%, but this is still lower than the results for a single gaze shift. Requiring more then two gaze shifts continued to lower the sensitivity to unacceptable levels.

These results seem to suggest at first that the gain of using an eye-tracker is not worth the cost of adding this information to the ACE model. However, there are a few counterarguments to this conclusion. First, it should be noted that time accuracy here is artificially high. One of the drawbacks of using time as a predictor of self-explanation is that the amount of time elapsed tells the model nothing about a student's behavior between actions. During a long time spent on a given case, a student may be doing or thinking of something completely unrelated to ACE. This seldom occurs in our data, as indicated by the high sensitivity of *time*, but it should be kept in mind that students were in a laboratory setting with little available distraction, in the presence of an observer and wearing a rather intrusive device, making it more difficult for the students' thoughts to wander and resulting in time being a more reliable indicator of self-explanation than it would be in actual practice.

Second, the sensitivity of the eye-tracker as a predictor may be artificially low due to errors in the eye-tracking data. Eye tracker calibration proved very difficult for participants with heavy eyelashes or thick glasses. The eye-tracker would function with a reading of "GOOD CALIBRATION" or "POOR CALIBRATION", and for several subjects "POOR CALIBRATION" was the best that could be achieved. Also, calibration could sometimes be compromised when a student sneezed or touched her face. Problems with calibration would make it more difficult for the eye-tracker to detect eye movements, and thus some gaze shifts could go unrecorded. These calibration problems are specific to a head-mounted device and would likely be less of an issue with a desk-mounted one. It should be noted that when we recomputed these accuracies using only the data points associated with the eleven students out of nineteen for whom "GOOD CALIBRATION" was achieved, sensitivity increased to 63.8%. Although this does not seem like a substantial increase, the reader should bear in mind that it is based on only 79 episodes (about 50% of the available data points) and possibly included students whose calibration was compromised during the interaction with ACE.

Finally, combining gaze shift and time into one predictor substantially improves sensitivity. That is, if an action is classified as self-explained when there is *either* a gaze shift *or* more than 16 seconds elapsed time, most of the self-explanation episodes (85.8%) are correctly recognized. This increase also causes the combined accuracy to improve.

However, as sensitivity increases with the combined OR predictor, specificity is reduced, such that only 62% of the episodes that lack self-explanation were discovered by the model. Here a tradeoff appears between sensitivity and specificity. Depending on how the system is used, it may be more important to correctly classify self-explanation when it occurs than to detect the lack thereof. This is the situation when letting natural self-explainers explore without interruption is given highest priority. Here, using the combination of the eye-tracker and time data is best. Alternatively, it may be more important to make sure that the system intervenes wherever it is necessary. Thus, failing to identify lack of self-explanation is a bigger problem than failing to detect it when it occurs. In this case, the eye-tracker alone is a more appropriate predictor, because students who need help will be more likely to get it.

There are also practical considerations. In some situations an eye-tracker might not be available due its cost or other factors. Then time, which is always simple and efficient to measure, would be the only predictor available. However, there may also be settings in which ACE users are surrounded

[7] Note that an alternative, possibly more intuitive way to combine the two predictors is to use an AND condition, to catch cases where a long time elapses because students are distracted, not because they are self-explaining. But the OR condition works better with our data because we have few such cases. Thus, elapsed time is indeed a good predictor for the presence of self-explanation, as we discuss later in the section.

[8] Note that we cannot report statistical significance on these results, as they represent individual numbers (percentages of cases classified correctly).

by such high levels of distraction (e.g., a noisy classroom) that time would perform exceptionally poorly. Then an eye-tracker would be preferable to present a much more reliable picture of the focus of the student's attention.

Given the above arguments, we felt that it is worthwhile adding eye-tracker data to the ACE model, and in such a way that allows for flexibility in deciding which predictor (or combination of predictors) to use, as we describe in the next section.

THE NEW ACE STUDENT MODEL

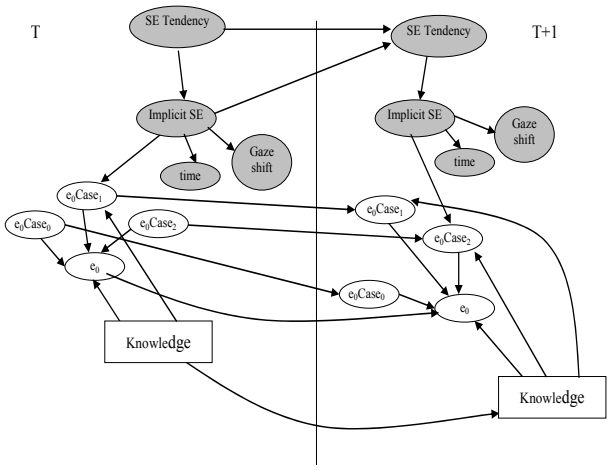

Figure 6. The revised ACE student model

Figure 6 shows the revised student model over two time slices, where the shaded nodes indicate the part of the model that we modified to include evidence from eye-tracking. In this model, after an exploratory action occurs (e.g., the action represented by the node e_0Case_1 in slice T), time is kept and eye movements are monitored until the next action. When the next action is carried out, a new slice is created; in parallel, if the new action is not an action indicating explicit self-explanation (i.e., a selection of predefined self-explanations in an interface menu), an *implicitSE* node is created for the previous action in slice T, along with *time, gaze shift* and *tendency-to-SE* nodes with the appropriate values. These new nodes are used to assess the effectiveness of the exploration case, updating the corresponding node, e_0Case_1, shown in Figure 6. This update further propagates to the exploration of exercise (node e_0 in Figure 6) and the other relevant exploration nodes shown in Figure 2. The second time slice shows the addition of analogous SE nodes after the student performs an exploratory action corresponding to the exploration case e_0Case_2

As Figure 6 shows, the revised model – termed *Gaze_Based model* from now on – relies on a clear separation between the causes of implicit self-explanation and its effects, i.e., gaze shifts and time on action. These effects are encoded as independent predictors, as in a naïve Bayesian classifier.

The main advantage to this approach is that it is highly modular, allowing the *gaze shift* and *time* nodes to be easily used or ignored as needed. Modularity, along with the fact that all the variables in this part of the model are observable, also facilitates learning the relevant conditional probabilities tables (CPTs, shown in Table 3) from frequencies in our dataset. For instance, the probability that there is a gaze shift if a student self-explains (first entry on the right in Table 3) is computed as the ratio of the number of data points showing both self-explanation and a gaze shift over the total number of data points corresponding to self-explanation episodes. Similarly, the probability that time on action is greater than 16 seconds if a student self-explains (first entry on the left in Table 3) is computed as the ratio of number of data points showing both self-explanation and time greater that 16 seconds over the total number of data points corresponding to self-explanation episodes. In the previous model, the portion that tracks self-explanation was based on intuition and reasonable estimates of conditional probabilities.

Table 3: CPTs for time and gaze shift in the new model

implicitSE	P(time < 16s)
Y	0.71
N	0.32

implicitSE	P(gaze shift)
Y	0.61
N	0.24

The disadvantage of this structure is that it assumes independence between time and the presence of gaze shifts, which is not necessarily true. In fact, our data actually suggests a small positive correlation between the two. However, similar assumptions in pure naïve Bayesian classifiers have been shown to perform surprisingly well in practice, even when this independence cannot be guaranteed.

Note that the version of the model in Figure 6 does not include the *Coach's hints to SE* nodes, nor the *Stimuli-to-SE* nodes shown in Figure 3. The *Coach's hints to SE* were removed because no hints were provided during the preliminary user study, and thus we had no data to set the relevant conditional probabilities. The *Stimuli-to-SE* nodes were removed because they were redundant, given that we were left with only one possible stimulus, the student's SE tendency.

To determine the relationship between tendency to self-explain and implicit self-explanation, the study participants were divided into *self-explainers* – those who self-explained at least 20% of the time – and *non-self-explainers* – those who did not. We found that self-explainers and non-self-explainers self-explained 79.8% and 13.3% of the time, respectively. These frequencies were then used to set the conditional probabilities for the Implicit SE node: the probability that there is a self-explanation episode if a student is a self-explainer was set to 0.8, while the probability that there is a self-explanation episode if the student is a non-self-explainer was set to 0.1

It should also be mentioned here that when we compared the average learning gain (difference from pre-test to post-test) of self-explainers and non-self-explainers, we found a mean 24% gain for the self-explainers, against a 5.7% for non-self-explainers. The difference was found to be statistically significant at the 0.05 level (two-tailed t-test), confirming that self-explanation has a significant effect on overall learning.

TESTING THE NEW STUDENT MODEL

In this section, the performance of the new, Gaze_Based, model, is evaluated using new user data. For purposes of comparison, we also tested the two previous versions of the ACE model: Action_Based model, which does not include self-explanation at all, and Time_Based model, with time only. This allows for an assessment of the incremental effects of adding self-explanation and then the gaze data to the ACE model.

In order to gain data for model testing, we ran 18 more subjects with the same experimental setup and data analysis adopted for the first study. As in the previous study, participants were university students who had not taken any college level math. This new set of subjects yielded 109 exploration cases with self-explanation and 68 without, which were then used to assess the performance of the three models.

In the reminder of this section, we first report the accuracy of the new model on this new data in assessing self-explanation (first subsection) and exploration (second subsection). We then describe a cross-validation analysis we carried out to provide a more precise picture of the overall model performance and stability in assessing individual students. In the last subsection, the model is tested using different evidence of implicit self-explanation.

Accuracy of Implicit SE Assessment

Table 4 Values of *implicitSE* nodes corresponding to actions in study data

action	Experts SE assessment	Time_Based model (time only)	Gaze_Based model (time and gaze shifts)
1	Y	0.698	0.723
2	N	0.287	0.180
3	Y	0.409	0.645

To test model accuracy in assessing implicit self-explanation, we needed a threshold probability to decide when an implicitSE node predicts the occurrence of self-explanation. We derived this probability from data from the previous study, as follows. Using a simulated student program, the log files from the first study (training data) were run through each of the two models that do assess implicit self-explanation, e.g., the Time_Based model and the Gaze_Based model. The probabilities of implicitSE nodes were then compared against the coded data points from the first study. Recall that each data point corresponds to a user action classified by human coders as self-explained or not. The implicitSE node in each model (see Figure 3 and Figure 6) also yielded probabilities that self-explanation occurred at the time of this action. These probabilities were compared to coder assessments to test the predictive performance of each model. A small fragment of this data appears in Table 4 .

To determine a good threshold over implicit SE nodes for each model, a Receiver Operating Characteristic (ROC) curve was constructed for these implicitSE probabilities. A ROC curve is a standard technique used in machine learning to evaluate the extent to which an information filtering system can successfully distinguish between relevant data (episodes the filter correctly classifies as positive, or true positives) and noise (episodes the filter incorrectly classifies as positive, or false positives), given a choice of different filtering thresholds.

Figure 7 shows the ROC curves for our two models, where the filter is the threshold over implicit SE probabilities. From these curves, we chose for each model the threshold that optimizes the tradeoff between true positive rate and false positive rate, as is standard practice in machine learning. These thresholds are marked by an asterisk in Figure 7.

Table 5 Accuracies of *implicitSE* nodes

	Time_Based model (time only)	Gaze_Based model (time and gaze shifts)
True Positive rate (*sensitivity*)	65.1%	71.6%
True Negative rate (*specificity*)	62.6%	74.3%
Combined	63.9%	73.0%

Next, the user log files from the new study (test data) were run through each model. Using the thresholds found from the training set, the model's implicitSE nodes were tested for accuracy against the new set of coded data. Table 5 shows the sensitivity, specificity and combined accuracy for the two models.

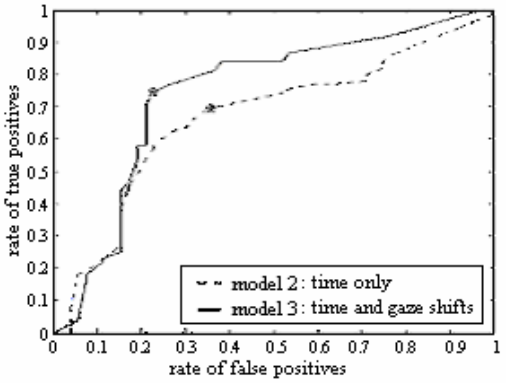

Figure 7. ROC curves for models as predictors of implicit self-explanation over training data

Here, the addition of the eye-tracker causes an increase in each of the measures, with the increase being more substantial for specificity. This is consistent with the assumption, supported by data in the first study, that the use of eye-tracking will catch many of the false positives inherent in the use of time as a predictor. To further compare the accuracy of the two models, we generated the ROC curves of their performance as predictors of implicit SE over the new data set (Figure 8).

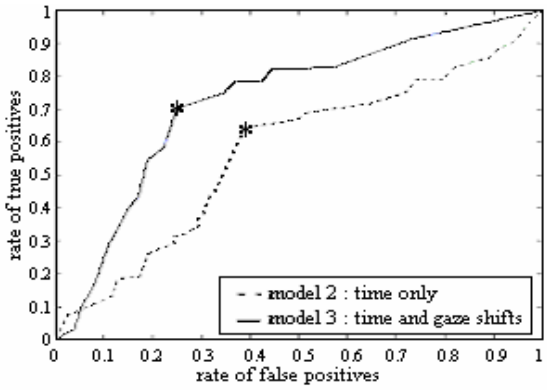

Figure 8. ROC curves for models as predictors of implicit self-explanation over testing data

The area under a ROC curve is equal to the probability that a randomly selected positive case will be given a higher probability by the model than a randomly selected negative case [14]. Thus, ROC curves with larger area correspond to better predictors over the data. As shown in Figure 8, the Gaze_Based model yields a ROC curve with greater area than that of Time_Based model. This difference in area is statistically significant to the $z > 1.96$ level [14].

Accuracy of Exploration Assessment

Each of the three versions of the ACE model (including the the Action_Based model, which does not assess self-explanation) was also evaluated as a predictor of adequate exploration. Data for this evaluation was collected as follows. In both user studies, participants completed a post-test on the mathematical concepts represented in the ACE model, immediately after interacting with ACE. A correspondence was then created between these concepts and specific post-test questions. These questions were then used to determine the student's aptitude in each of the concepts (e.g., positive intercepts and negative intercepts) at the end of the experiment. In addition, when a student's log file is run through any of the three models, the final probabilities of the exploration of categories nodes (e.g., nodes Negative Intercept and Positive Intercept in Figure 2) represent the model assessment that the student understands these concepts at the end of the interaction (i.e., that she adequately explored this material). This assessment can then be compared with the corresponding post-test scores to evaluate model accuracy over effective exploration.

Table 6: Accuracies of exploration nodes

	Action_Based model	**Time_Based Model (time only)**	**Gaze_Based model (time and gaze shifts)**
True Positive rate (*sensitivity*)	62.7%	70.4%	73.9%
True Negative rate (*specificity*)	55.2%	71.5%	76.3%
Combined	59.0%	71.0%	75.1%

As before, a ROC curve was constructed for each model over the training data to determine the best threshold at which an exploration node could predict adequate exploration – and thus understanding – of the material. These thresholds were then used to determine the accuracy of each model over the testing data, resulting in the accuracies listed in Table 6. Each of the accuracies increased with each successive model, indicating that the addition of self-explanation and gaze shift data yielded improvements. It also confirms that an increase in the accuracy of implicit self-explanation detection does in fact cause an increase in the accuracy of exploration assessment.

ROC curves were also generated to compare each model's performance on exploration assessment over the test set; these appear in Figure 9. As shown in the figure, the area under the curve increased with the inclusion of self-explanation in the student model. The addition of gaze-shift data also caused an increase. Both of these increases were found to be statistically significant at the $z > 1.96$ level [14].

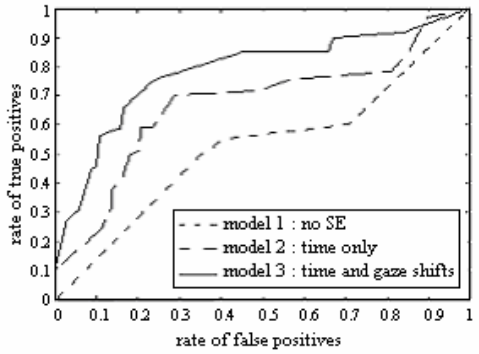

Figure 9. ROC curves for models as predictors of sufficient exploration over testing data

It should, however, be noted that the increase in accuracy caused by the addition of the eye-tracker is higher for the implicitSE nodes than for the exploration nodes. This is due to the difference in the way each is measured. Each implicitSE probability is taken at the time that the associated action occurs while only the probabilities of the exploration nodes at the end of the interaction are used in the analysis. Thus, while the implicitSE nodes represent the state of the user at a specific time and are strongly affected by the presence or absence of a gaze shift, the exploration nodes' final probabilities are the result of many actions throughout the interaction and are influenced by other factors. Given these results, we can conclude that the main benefit in adding eye-tracking versus using time only is the more accurate assessment of implicit self-explanation, which allows ACE to generate more precise real-time interventions during the student's interaction with the system.

Cross-validation Analysis

We conducted a cross-validation analysis to get a better picture of how the various models performed on individual students. For each student model, leave-one-out cross-validation was performed using all 36 students from both studies. This involved isolating a student and then setting model thresholds and conditional probabilities using the data from all remaining students. This procedure was performed for each of the 36 students, and the accuracy results from each student were averaged.

We started the analysis by assigning a generic prior probability of 0.5 to knowledge and tendency to SE nodes in each set. The mean combined accuracies and the standard deviations for the implicitSE nodes for each model with generic priors are given in the first row of

Table 7: Cross-Validation results for ImplicitSE nodes for different models and priors

		Time_Based model	Gaze_Based model
Generic priors	**Combined Accuracy**	62.1%	71.6%
	St, Dev.	8.1%	7.9%
Customized Knowledge Node Priors	**Combined Accuracy**	65.8%	75.2%
	St, Dev.	7.4%	7.2%
Customized Knowledge and SE Tendency Node Priors	**Combined Accuracy**	67.2%	76.4%
	St, Dev.	7.6%	7.1%

Table 8: Cross-validation results for ImplicitSE nodes for different models and priors

		Action_Based model	Time_Based model	Gaze_Based model
Generic priors	**Combined Accuracy**	57.3%	65.3%	71.6%
	St. Dev.	11.6%	9.4%	8.7%
Customized Knowledge Node Priors	**Combined Accuracy**	64.7%	69.9%	76.8%
	St. Dev.	10.1%	9.3%	8.4%
Customized Knowledge and SE Tendency Node Priors	**Combined Accuracy**	68.4%	70.4%	77.5%
	St. Dev.	9.2%	10.3%	7.9%

Table 7. These values show improved performance with the addition of the eye-tracker as well as slightly higher stability. The performance difference is statistically significant at the 0.05 level (one-tailed t-test).

The first row of Table 8 shows analogous results for the exploration nodes. As before, there is an improvement in mean accuracy with each successive model. ANOVA analysis showed statistical significance in the differences within the set of mean accuracies, and a one-tailed T-tests showed that the differences between each model are statistically significant.

We then looked at the influence of assigning student tailored priors on each model's performance. Knowledge node priors were set based on each participant's performance on the study pre-test. If the student answered the corresponding pre-test items correctly, a prior probability of 0.85 was assigned to the corresponding knowledge node. Otherwise, the probability was set to 0.15. These values were chosen as reasonable estimates since they are close to 1 and 0, but they still allow for those students who guess correctly or make errors even though they understand the concept. Also, early analysis showed that the model is not sensitive to small changes in these prior probabilities (e.g., using 0.9 and 0.1 instead).

Cross-validation was then performed again using customized prior probabilities for knowledge nodes. The mean accuracies for the implicitSE nodes appear in the second row of Table 7. While customizing these prior probabilities causes an increase in accuracy and stability for each model, this increase was statistically significant only for the model that uses time and gaze shifts to detect self-explanation. Results with customized priors are also given for the exploration nodes in the second row of Table 8. Here, the customization causes a statistically significant increase in the mean accuracy for each model, as well as an increase in stability.

Priors for the tendency to SE node were derived from our previously discussed classification of study participants into self-explainers – those who self-explained at least 20% of the time – and non-self-explainers – those who did not. If a student was classified as a self-explainer, the prior probability for her tendency to SE node was set to 0.85, while for a non-self-explainer, a value of 0.15 was used (these values were arbitrarily picked after trying a few for both the high and low probabilities and realizing that the model was not sensitive to small changes over them). Repeating the cross-validation procedure using tailored priors for both knowledge and Tendency to SE nodes yielded the results given in the third rows of Table 7 and Table 8. In each case, for each model the improvement brought about by the customization of the Tendency to SE node failed to achieve statistical significance, showing that the model is not very sensitive to this parameter. However, we believe it is still worth keeping this node in the model, for two reasons. First, it provides ACE with an extra piece of information on potential causes of poor exploration by a student (i.e., low self-explanation tendency). Second, its influence may become more relevant in the presence of the "coach hint to self-explain" node, which we plan to add as an additional cause of implicit self-explanation once we add data on the effect of these hints on student behavior.

In summary, we found that adding eye-tracking to the student model causes a statistically significant improvement in the assessment of both implicit self-explanation and sufficient exploration. It is also advantageous to use pre-test results, if available, to customize the prior probabilities of the knowledge nodes. Tailoring the tendency to SE prior probabilities, however, fails to bring about a significant improvement.

Performance with Different Evidence

This section illustrates how the new model's (Gaze_Based model) performance changes depending upon the type of evidence used (time alone, gaze shifts alone or both).

The log files of the new study participants were run through the new model two more times, once withholding eye-tracking data, once withholding time data. For each run, the accuracy of the model's assessment over implicit self-explanation and exploration were computed as described earlier, yielding the results in Table 9. For purposes of comparison, the table also repeats the accuracies of the model that receives evidence from both time and gaze shifts. As shown in the table, information on time alone generates higher sensitivity than information on gaze shifts alone, while the latter generates higher specificity. These findings match those of the original user study [9]. They are also consistent with the assumption that time overestimates

self-explanation behavior by assuming that the user spends all idle time considering the exploration. For each measure, the combined predictor outperforms either on its own

.

Table 9: *ImplicitSE* accuracies for ACE Gaze_Based model using different predictors as evidence of implicit self-explanation

	Time evidence only	Eye-tracking evidence only	Time and eye-tracking evidence
True positive rate (sensitivity)	67.9%	62.3%	71.6%
True negative rate (specificity)	64.8%	67.8%	74.3%
Combined	66.3%	65.1%	73.0%

A similar analysis was performed to assess the influence of evidence type over exploration assessment, with results reported in Table 10. As with the implicitSE nodes, information on time alone has a higher sensitivity than using only gaze shifts. However, gaze shifts alone achieve higher specificity. These predictors combine to yield the highest accuracy for each measure. This is due to the fact that accuracy improves with more evidence used. It should also be noted that each single predictor seems to succeed where the other fails, so this complementary behavior likely contributes to the high accuracy of the combined predictor.

Table 10: *Exploration* accuracies for ACE Gaze_Based model using different predictors as evidence of implicit self-explanation

	Time evidence only	Eye-tracking evidence only	Time and eye-tracking evidence
True positive rate (sensitivity)	71.2%	69.8%	73.9%
True negative rate (specificity)	72.9%	73.4%	76.3%
Combined	72.1%	71.6%	75.1%

Notably, the accuracies generated by the new model when only time information is used are comparable to (although slightly higher than) the accuracies of the Time_Based model, despite the differences in structure and method of CPT definition (data-based for the Gaze_Based model and expert-based for the Time_Based model).

Pupil Dilation as Predictor of Self-explanation

In the previous sections, we showed that gaze-pattern information as detected by an eye-tracker can improve real-time assessment of user self-explanation, and consequent exploration behavior, during interaction with an ELE. The eye-tracker we used also records user pupil size during fixations. Pupil size has been shown to have a positive correlation with cognitive load. Since self-explanation requires cognitive effort that may increase a user's cognitive load, we wanted to check whether we could use pupil size as an additional means of detecting self-explanation in the ACE student model. In this section, we describe the results of this investigation.

Data collection

In the user studies described in previous sections, the eye-tracker generated a log file containing, in addition to gaze data, the diameter of the user's pupil throughout the interaction.

Several factors are known to influence pupil size in addition to cognitive load [4], including ambient lighting, the size of the eye itself and even nonvisual stimuli such as sound. Since it is not possible to keep environmental conditions adequately constant from one study session to the next or to ensure that users have similar pupils, it is common practice to use normalization to get a pure correlation between pupil size and cognition. One standard normalization method [28] involves taking a baseline measurement at a time when all users should have the same cognitive state. This baseline encapsulates all information regarding each user's pupil attributes, as well as environmental conditions. When the baseline is subtracted from all other pupil size values it yields a normalized measure that can then be compared across users.

We collected baseline measurements for all 18 subjects in the second ACE study. Participants were asked to turn off the computer monitor and stare at the blank screen for a few seconds after completing their interaction with ACE. This baseline was chosen because it was assumed that sitting idle while staring at the same visual stimulus at the close of the interaction with ACE would bring the participants to the same cognitive state.

Recall that the second user study resulted in a set of exploration cases in which observers determined the presence or absence of self-explanation. For each of these data points, we computed the user's average pupil size after the action but before the next action[9]. This was carried out as follows. During the time interval between actions, the learner made a series of eye fixations recorded by the eye-tracker. In addition to location coordinates, the data for each fixation included the length of the fixation in milliseconds and the user's pupil size (given as image area in pixels) when it occurred. For each of these fixations, the pupil size and duration were multiplied. These values were then summed and divided by the total elapsed time, resulting in a weighted average pupil size between actions. Finally, the baseline measurement was subtracted, resulting in a normalized value. Table 11 shows a sample of this data. Note that each normalized value in the table is negative. While one might expect that users had a low cognitive load – and thus small pupil size – during baseline recording, this could not be guaranteed. For baseline purposes, it was only necessary that students be in the same cognitive state. Further, students would naturally have larger pupils when staring at a blank monitor than when looking at a lit screen.

After determining a normalized average pupil size value for each of the available data points, we checked whether there was any difference in the pupil size when users were self-explaining and when they were not. We found that users had a mean normalized pupil size of -56 when they self-explained and

[9] We actually excluded from the original data set those points in which self-explanation did not happen immediately after the action, to simplify data analysis.

-59 when they did not. However, due to the large standard deviations (14.4 and 10.5, respectively), this difference fails to achieve statistical significance (as measured by a two-tailed T-test).

A ROC curve was also created for these points to test the performance of pupil size as a predictor of self-explanation for different normalized pupil size thresholds. This curve appears in Figure 10. The area under the curve is 0.43, less than the area of 0.5 given by random chance. Thus, over this data, pupil size is definitely not an acceptable predictor of self-explanation.

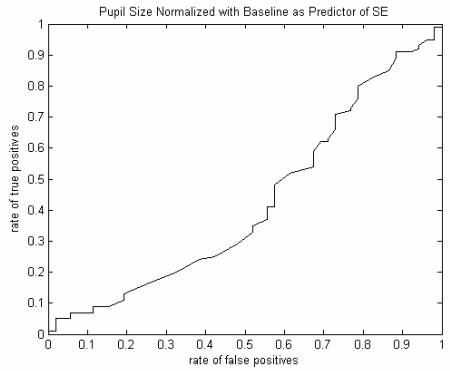

Figure 10: ROC curve for pupil size as a predictor of self-explanation

One possible cause for this result may relate to the assumption that all users who stared at the blank screen after interacting with ACE were in approximately the same cognitive state. This might not have been the case, generating an inappropriate baseline. To test this theory, we tried a different data normalization technique, i.e., we used Z-scores to normalize pupil sizes among participants. Each mean pupil size was normalized according to the following formula:

$$X_k' = \frac{X_k - m}{s}$$

where X_k' is the normalized mean pupil size after action k, X_k is the mean pupil size after action k (see values in the third column of Table 11), m is the average value of X_k over all actions by the user, and s is the standard deviation. While the mean pupil size m encapsulates all information that affects the user's pupil size (i.e., lighting conditions and eye size), it is not specific to any single moment during the experiment and thus does not require the assumption that we identify the point at which all users are in the same state, as is required for the baseline method.

Table 11: Sample fragment of pupil size data

User #	Self-Expl.?	Mean pupil size after action	User baseline	Normalized value
7	Y	1321	1391	-70
7	Y	1359	1391	-32
8	N	550	606	-56
8	Y	584	606	-22
…	…	…	…	…

As before, we determined whether there was a significant difference in the pupil size of users who were self-explaining and those who were not. Users had a mean normalized pupil size of 0.928 when they self-explained and 0.812 when they did not. However, due to the large standard deviations (0.37 and 0.29, respectively) this difference also fails to achieve statistical significance. The ROC curve for pupil size as a predictor of self-explanation for different Z-score normalized pupil size thresholds has an underlying area of 0.49, about the same as that given by random chance. This confirms that, even with Z-score normalization, pupil size is not an acceptable predictor of self-explanation.

We then hypothesized that the negative results may be due to incorrectly assuming that the effect of self-explanation on cognitive load would span the whole interval between the self-explained action and the following one. One possibility is that positive self-explanation caused the greatest increase in cognitive load – and thus pupil size – immediately after the outcome of the corresponding exploratory action. In this case, examining the weighted mean pupil size over the whole time interval between actions would fail to capture this behavior, as the pupil size increase would be dissipated by the smaller pupil size values at the end of the time interval. Figure 11(a) shows a plot of the hypothetical pupil size in this case. Another possibility is that the greatest increase in cognitive load occurred at the end of the time interval, after the user had noticed the action outcome and had time to consider its meaning. These two cases appear together in Figure 11 (b). In either case, taking the mean pupil size over only the middle portion of the interval, as shown in the figure, would best capture the pupil size increase.

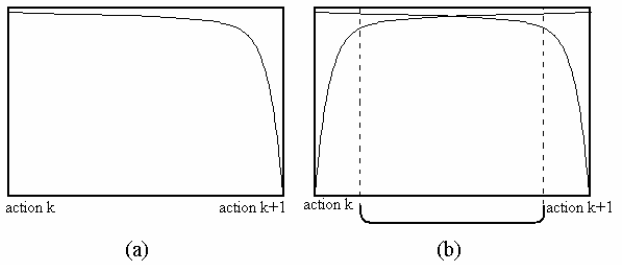

Figure 11: Hypothetical pupil size plotted over the time interval when it is largest at (a) the beginning and (b) at the beginning or the end (b). In (b), dotted lines indicate the middle 60% of the time interval

To test this theory, we calculated normalized weighted pupil sizes (using Z-score normalization) over only the middle 60% of each time interval. With this new measure, users had a mean normalized pupil size of 0.861 when self-explaining and a mean pupil size of 0.794 otherwise. The difference of these means still failed to reach statistical significance. The area under the ROC curve for this data was found to be 0.41, again less than that which would have resulted from random chance.

Discussion

The results discussed in the above section provide an initial indication that pupil dilation may not be a good predictor of self-explanation. This could be due to two reasons. The first is the inadequacy of pupil dilation as a detector of the differences in cognitive loads generated by presence or absence of self-explanation. The second is that such differences are not substantial enough to be detectable. Our data does not provide conclusive evidence to discriminate between the two explanations, nor are we aware of any cognitive science research on the relationship between self-explanation and cognitive load that may provide insights into this issue. However, the large standard deviations we found in all of the different cognitive load measures we tried speak in favor of the second hypothesis. That is, there seem to be cases when users were not self-explaining but their cognitive load was relatively high, perhaps because of the very fact that they were interacting with a complex system or thinking about mathematical functions. The rather intrusive eye-tracker may have also interfered with some of the subjects' thinking, thus working as an external cause of increased cognitive load. It is also possible that for some of the study participants, the spontaneous self-explainers in particular, self-explanation comes quite naturally and thus does not involve an increase in cognitive load significant enough to be reflected in their pupil size.

Note that support for the first hypothesis, i.e., that pupil size is not always a good predictor of cognitive load, may come from the findings in [27]. This work described a study that failed to show a correlation between pupil size and the difficulty of text subjects were asked to read. One could conclude from these results that pupil size is not a good predictor of cognitive load during reading tasks if the reading tasks in the study did indeed cause detectable cognitive load differences. This was not necessarily the case, however. The different texts used were classified as easy or difficult overall but, being quite long, may have had passages of varying difficulty. Thus, subjects' cognitive load may have significantly fluctuated within each reading task, making it impossible to detect any significant differences when the load was measured over the complete task.

DISCUSSION AND FUTURE WORK

In this paper, we have presented research on using real-time eye-tracking data for the on-line modeling of user self-explanation and effective exploration, two meta-cognitive behaviors relevant for student learning during interaction with ACE, an exploratory learning environment for mathematical functions. The goal is to enable the environment to provide adaptive support to improve these meta-cognitive behaviors and consequent student learning.

The main contribution of the paper is a formal evaluation showing that the model including eye-tracking information on user gaze shifts provides a more accurate assessment of student self-explanation than a model using only time as a lower-level predictor. Our evaluation also shows that modeling self-explanation improves the assessment of student exploratory behavior, as opposed to relying only on student interface actions. This result supports the argument that modeling high-level user traits can improve the adaptive capability of an Intelligent User Interface, providing an initial justification for the effort involved in this type of sophisticated user modeling.

One of the avenues of future work in this research is to further explore gaze-tracking as a predictor of self-explanation and effective exploration by considering additional gaze patterns in addition to the gaze shifts discussed in this paper. We have started to investigate the distinction between direct and indirect gaze shifts by using unsupervised machine learning (clustering) on these two gaze patterns in conjunction with all ACE interface actions (not only plot and equation changes considered in this paper). This enabled us to identify clusters of students with similar interaction behaviors and learning outcomes [2]. While intuition suggests that indirect gaze shifts may indicate student distraction, our preliminary results show that they are a better predictor of effective exploration than direct gaze shifts. Although our finding that direct gaze shifts seem to be less informative than indirect ones may be an artifact of the available data points, the fact that indirect gaze shifts do correlate with effective exploration shows that they generally do not indicate distraction, consistent with what we had assumed in the research presented here. However, these findings are based on the data set collected in the constrained laboratory setting described in this paper. It will be interesting to investigate if and how they transfer to a more natural, noisy environment where it is easier for a student to be distracted.

Another question that we are planning to address in future research is whether gaze patterns other than gaze shifts can predict effective exploration. For instance, what if the student does not shift between the graph and plot regions, but looks at one of them for a while? Can this indicate self-explanation? Since the space of potentially meaningful patterns can be rather large and some of them may be unintuitive, we are planning to use unsupervised clustering to mine gaze data for these patterns.

In this paper we have also presented results on the performance of pupil dilation as a predictor of self-explanation. The rationale behind investigating this predictor relies on the link that has been found between pupil size and cognitive load, and on the assumption that self-explanation may generate detectably higher cognitive load than lack of it. Our findings, however, suggest that pupil size is not a reliable predictor of self-explanation, at least during interaction with ACE.

The final proof of the utility of rich user models must come from empirical evidence that adaptive intelligent interfaces based on these models improve user performance. The next step in our research is to provide this empirical evidence for ACE. We have designed a variety of interface tools that allow ACE to provide different levels of prompting for both exploration and self-explanation by relying on the assessment of the student model described here. We are in the process of designing a user study to test the effectiveness of these adaptive tools. One of the goals of the study will be to gain better insight on the distinction between explicit and implicit self-explanation, in particular on their relative frequency and on how much is gained by being able to detect implicit over explicit self-explanation, which is significantly easier to detect.

A longer-term research avenue is to investigate how the results presented in this paper generalize to other tasks and domains. For instance, Conati and Vanlehn [10] proposed a system that tracks user self-explanation of examples during problem solving of physics problems by making each individual example line visible only if the student moves the mouse over it. This allowed the system to gather data on the

latency of student attention for each uncovered line, which was then used as a prediction of self-explanation in the system's user model. We are planning to substitute the mouse-based tracking mechanism with an eye-tracker to ascertain how much more information it can provide on useful user gaze patterns, as well as to compare the two interfaces in terms of user disruption.

AKNOWLEDGENTS
We thank the National Science and Engineering Research Council of Canada (NSERC) for funding this research. We also would like to thank David Ternes for writing the algorithm that parses raw eye-tracker data, Kasia Muldner for helping with the user studies, and Giuseppe Carenini for his insights on ROC curve analysis.

REFERENCES

1. V. Aleven and K.R. Koedinger, An Effective Meta-Cognitive Strategy: Learning by Doing and Explaining with a Computer-Based Cognitive Tutor. Cognitive Science 2 (2) (2002) 147-179.
2. S. Amershi and C. Conati, Unsupervised and Supervised Machine Learning in User Modeling for Intelligent Learning Environments, in: Intelligent User Interfaces, (2007).
3. M. Bassok, Transfer of Domain-Specific Problem-Solving Procedures. Journal of Experimental Psychology: Learning, Memory and Cognition 16 (3) (1990) 522-533.
4. J. Beatty, Task-Evoked Pupillary Responses, Processing Load, and the Structure of Processing Resources. Psychological Bulletin 91 (1982) 276-292.
5. A. Bunt and C. Conati, Probabilistic Student Modeling to Improve Exploratory Behaviour. Journal of User Modeling and Usre-Adapted Interaction 13 (3) (2003) 269-309.
6. A. Bunt, C. Conati and K. Muldner, Scaffolding Self-Explanation to Improve Learning in Exploratory Learning Environments. 7th International Conference on Inteligent Tutoring Systems (2004).
7. M.T.H. Chi, Constructing Self-Explanations and Scaffolded Explanations in Tutoring. Applied Cognitive Psychology 10 (1996) S33-S49.
8. M.T.H. Chi, M. Bassok, M.W. Lewis, P. Reimann and R. Glaser, Self-Explanations: How Students Study and Use Examples in Learning to Solve Problems. Cognitive Science 15 (1989) 145-182.
9. C. Conati, C. Merten, K. Muldner and D. Ternes, Exploring Eye Tracking to Increase Bandwidth in User Modeling, in: 10th Annual Conference on User Modeling, (2005).
10. C. Conati and K. Vanlehn, Toward Computer-Based Support of Meta-Cognitive Skills: A Computational Framework to Coach Self-Explanation. International Journal of Artificial Intelligence in Education 11 (2000).
11. T. De Jong and R. Van Joolingen, Scientific Discovery Learning with Computer Simulations of Conceptual Domains. Review of Educational Research 68 (179-201) (1998).
12. T. Dean and K. Kanazawa, A Model for Reasoning About Persistence and Causation. Computational Intelligence 5 (3) (1989) 142-150.
13. D. Decarlo and A. Santella, Stylization and Abstraction of Photographs, in: 29th Annual Conference on Computer Graphics and Interactive Techniques, (2002) 769-776.
14. J. Fogarty, R. Baker and S. Hudson, Case Studies in the Use of Roc Curve Analysis for Sensor-Based Estimates in Human-Computer Interaction, in: Graphics Interface, (2005) 129-136.
15. K.A. Gluck and J.R. Anderson, What Role Do Cognitive Architectures Play in Intelligent Tutoring Systems?, in Cognition and Instruction: Twenty-Five Years of Progress, D. Klahr and S.M. Carver, EditorsErlbaum, 2001) 227-262.
16. A.J. Hornof, A. Cavender and R. Hoselton, Eyedraw: A System for Drawing Pictures with Eye Movements, in: ASSETS 2004" The Sixth International ACM SIGACCESS Conference on Computers and Accessibility, (2004) 86-93.
17. S.T. Iqbal and B.P. Bailey, Using Eye Gaze Patterns to Identify User Tasks. The Grace Hopper Celebration of Women in Computing (2004).
18. S.T. Iqbal, X.S. Zheng and B.P. Bailey, Task-Evoked Pupillary Response to Mental Workload in Human-Computer Interaction, in: ACM Conference on Human Factors in Computing Systems, (2004) 1477-1480.
19. R. Jakob, The Use of Eye Movements in Human-Computer Interaction Techniques: What You Look at Is What You Get, in: (1998) 65-83.
20. P. Majaranta, A. Aula and K.-J. Raiha, Effects of Feedback on Eye Typing with a Short Dwell Time, in: Symposium on Eyetracking Research and Applications, (2004) 139-146.
21. T. Mitrovic, Supporting Self-Explanation in a Data Normalization Tutor. Supplementary Proceedings of AIED2003 (2003).
22. R.A. Monty and J.W. Senders, eds. Eye Movements and Psychological Processes. 1976, Lawrence Erlbaum Associates: Hillsdale, New Jersey.
23. M. Njoo and T.D. Jong, Exploratory Learning with a Computer Simulation for Control Theory: Learning Processes and Instructional Support. Journal of Research in Science Teaching 30 (8) (1993) 821-844.
24. L. Qu and L. Johnson, Detecting the Learner's Motivational States in an Interactive Learning Environment, in: 12th International Conference on Artificial Intelligence in Education, (2005).
25. A. Renkl, Learning Mathematics from Worked-out Examples: Analyzing and Fostering Self-Explanation. European Journal of Psychology and Education in press (1999).
26. D. Salvucci and J. Anderson, Intelligent Gaze-Added Interfaces, in: SIGHCI Conference on Human Factors in Computing Systems, (2000).
27. M. Schiessl, S. Duda, A. Tholke and R. Fischer, Eye Tracking and Its Application in Usability and Media Research. "Sonderheft: Blickbewegung" in MMI-interaktiv Journal 6 (2003).
28. H. Schultheis and A. Jameson, Assessing Cognitive Load in Adaptive Hypermedia Systems: Physiological and Behavioral Methods, in: Adaptive Hypermedia, (2004) 225-234.
29. V.J. Shute, A Comparison of Learning Environments: All That Glitters... in Computers as Cognitive Tools, S. Lajoie, P. and S. Derry, Editors. (Lawrence Erlbaum Associates, Hillsdale, NJ, 1993) 47-73.

30. V.J. Shute and R. Glaser, A Large-Scale Evaluation of an Intelligent Discovery World. Interactive Learning Environments 1 (1990) 51-76.
31. J.L. Sibert, M. Gokturk and R.A. Lavine, The Reading Assistant: Eye Gaze Triggered Auditory Prompting for Reading Remediation, in: 13th Annual ACM Symposium on User Interface Software and Technology, (2000).
32. I. Starker and R.A. Bolt, A Gaze-Responsive Self-Disclosing Display. CHI: Human Factors in Computing Systems (1990).