Gaze Analysis methods for Learning Analytics

THÈSE N^{O} 6696 (2015)

PRÉSENTÉE LE 6 NOVEMBRE 2015

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS
LABORATOIRE D'ERGONOMIE ÉDUCATIVE
PROGRAMME DOCTORAL EN INFORMATIQUE ET COMMUNICATIONS

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Kshitij SHARMA

acceptée sur proposition du jury:

Prof. P. Fua, président du jury
Prof. P. Dillenbourg, Dr P. Jermann, directeurs de thèse
Dr D. Gergle, rapporteur
Prof. T. van Gog, rapporteuse
Prof. A. Billard, rapporteuse

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2015

Not everything that can be counted counts,
and not everything that counts can be counted
- Albert Einstien

To Geeta, Hridai, and Nisha . . .

Acknowledgements

This dissertation is a result of the inspirations, efforts, and contributions of many people, who I have worked with and to whom I owe my deepest gratitude. First of all, I would like to thank my thesis advisors, Prof. Pierre Dillenbourg and Dr. Patrick Jermann. My thesis owes its existence to Pierre and Patrick. They both portray the definition of a teacher by Albert Einstein- *"It is the supreme art of the teacher to awaken joy in creative expression and knowledge"*. Their continuous guidance and encouragement kept me motivated to work hard everyday for the last four years. In our meetings, I always felt *"The older I got, the smarter my teachers became"*. -Ally Carter, *Out of Sight, Out of Time*. Their constructive critique helped me expanding my knowledge and limits. Without their scientific advice and stimulating ideas this research work would have never reached this state of maturity.

I would like to extend my gratitude to the members of my review committee: Prof. Darre Gergle, Prof. Tamara van Gog, and Prof. Aude Billard. I owe them a heartfelt appreciation for their constructive remarks on my dissertation. Their insightful comments helped me a lot to improve my thesis.

During different experiments, there were many people helping me understand, design, and conduct the experiments. I would like to thank Marc-Antoine Nüssli for carrying out the pair programming experiment and helping me understand various technical details about the experiment during my early days in CRAFT. I am grateful to Daniela Caballero Díaz and Himanshu Verma for their support in carrying out the Dual eye-tracking experiment with MOOCs. Without their support I would not have been able to complete the experiment in the given time limit. I also found encouragement from collaborating with Prof. Jérôme Chenal, who let me experiment with his MOOC. He was helpful and gave me useful ideas during the various parts of the experiment. He was also patient enough to use the eye-tracking glasses for multiple testing sessions. Without his help the experiment with displaying teacher's gaze on a MOOC would not have been realised.

As James Baldwin said in *Giovanni's Room- "Perhaps home is not a place but simply an irrevocable condition"*. I was lucky enough to find friends that made James Baldwin true for my PhD life. Life would have been very difficult in Lausanne had't I met people to have beer with. These were the people with whom I had uncountable beer talks, numerous coffee room discussions, many ping-pong sessions, and nice dinners- everything a person can find in a “irrevocable condition”. Sharing the workspace with these people was actually fun for me. I heartily appreciate Himanshu Verma, Hamed Alavi, Quentin Bonnard, Andrea Mazzei, Sébastien Cuendet, Frédéric Kaplan, Son Do Lenh, Tabea Koll, Sophia Schwär, Julia Fink, Nan Li, Olivier Guédat, Mirko Raca, Daniela Caballero Díaz, Afroditi Skevi, Lorenzo Lucignano, Sévéin Lemaignan, Engin Bumbacher, Nikolaos Maris, Valerie Bauwens, Luis Pablo Prieto Santos, Łukasz Kidzinski, Mina Shirvani, Ayberk Ozgur, Ashish Ranjan Jha, Lukas Hostettler for their friendship from the beginning of my work at CRAFT and later at CHILI.

Next, I would like express my gratitude to the people who made my life in Lausanne even more worth to leave India and come here. I would like to thank Jessica Delher and Guillaume Zufferey, and Jo’an and Guillaume Bardy, Farzaneh Bahrami, Huong-Ly Mai, Lorin Cuendet, Sonja Raca, María Jesús Rodríguez-Triana, Cristián Mansilla for having nice evenings over beer, food, crazy discussions and dance. I would also like to thank Loreline and Benno Zufferey, Miriam Cuendet, Léanne Bardy, Moud and Zoey Lemaignan, Dimitrije Raca, my tiny friends, who always made me happy with their cute smiles.

Himanshu Verma, I am thankful to him for helping me settle down in Lausanne during my early days, for telling me how to deal within the Swiss offices, banks, post-offices and EPFL HR office; and also for being an awesome flatmate, and for constantly reminding me to keep my room clean.

I am also indebted to all the “anonymous” participants of the studies, for their sincere, pretensionless, and genuine participation; even though I was forcing them to sit straight with their chins held up by an uncomfortable ophthalmologic rest. I would also like to thank National Science Foundation (NSF) for their support during my thesis.

This acknowledgement can never be complete without mentioning the “real boss” of the team: Florence Colomb. I would like to thank her for efficiently and effectively taking care of every administrative processes, sharing my frustration over visa issues, and taking care of my different names in different offices due to the absence of my surname.

I could never express my gratitude for my parents, Geeta Sharma and Hridai Narayan Sharma, for their love, care, and support. They are and will always be a motivating factor for me to be able to make them proud. I hope this thesis will add a bit to that goal. Enfin, I would like to thank the love of my life, Nisha. She was always there for me. I really cannot thank her enough for coping up with my crazy work schedule, completely changing her life for being close to me, her unfailing love, and unconditional support throughout this adventure. I dedicate my thesis to Nisha.

Lausanne, 13 October 2015 Kshitij Sharma.

Abstract

Eye-tracking had been shown to be predictive of expertise, task-based success, task-difficulty, and the strategies involved in problem solving, both in the individual and collaborative settings. In learning analytics, eye-tracking could be used as a powerful tool, not only to differentiate between the levels of expertise and task-outcome, but also to give constructive feedback to the users. In this dissertation, we show how eye-tracking could prove to be useful to understand the cognitive processes underlying dyadic interaction; in two contexts: pair program comprehension and learning with a Massive Open Online Course (MOOC). The first context is a typical collaborative work scenario, while the second is a special case of dyadic interaction namely the teacher-student pair.

We also demonstrate, using one example experiment, how the findings about the relation between the learning outcome in MOOCs and the students' gaze patterns can be leveraged to design a feedback tool to improve the students' learning outcome and their attention levels while learning through a MOOC video. We also show that the gaze can also be used as a cue to resolve the teachers' verbal references in a MOOC video; and this way we can improve the learning experiences of the MOOC students.

This thesis is comprised of five studies. The first study, contextualised within a collaborative setting, where the collaborating partners tried to understand the given program. In this study, we examine the relationship among the gaze patterns of the partners, their dialogues and the levels of understanding that the pair attained at the end of the task.

The next four studies are contextualised within the MOOC environment. The first MOOC study explores the relationship between the students' performance and their attention level. The second MOOC study, which is a dual eye-tracking study, examines the relation between the individual and collaborative gaze patterns and their relation with the learning outcome. This study also explores the idea of activating students' knowledge, prior to receiving any learning material, and the effect of different ways to activate the students' knowledge on their gaze patterns and their learning outcomes.

The third MOOC study, during which we designed a feedback tool based on the results of the first two MOOC studies, demonstrates that the variables we proposed to measure the students' attention, could be leveraged upon to provide feedback about their gaze patterns. We also show that using this feedback tool improves the students' learning outcome and their attention levels.

The fourth and final MOOC study shows that augmenting a MOOC video with the teacher's gaze information helps improving the learning experiences of the students. When the teacher's

gaze is displayed the perceived difficulty of the content decreases significantly as compared to the moments when there is no gaze augmentation.
In a nutshell, through this dissertation, we show that the gaze can be used to understand, support and improve the dyadic interaction, in order to increase the chances of achieving a higher level of task-based success.

Key words: Eye-tracking, Dual eye-tracking, Massive Open Online Courses (MOOCs), Learning analytics, Program comprehension, Pair programming, Collaborative problem solving.

Résumé

L’oculométrie est un moyen de prédiction de l’expertise, et de la performance de résolution de tâches. Les traces occulaires reflètent également la difficulté de la tâche et les stratégies impliquées dans la résolution de problèmes, aussi bien en mode individuel qu’en mode collaboratif. En analyse de l’apprentissage, L’oculométrie peut être utilisée comme un outil pour différencier les niveaux d’expertise et de résultats ainsi que pour offrir un feedback constructif aux utilisateurs. Dans cette thèse, nous montrons comment L’oculométrie contribue à la compréhension des processus cognitifs sous-jacents à l’interaction en binôme dans deux contextes : la compréhension d’un programme informatique à deux et l’apprentissage avec un MOOC (Massive Open Online Course ou cours en ligne ouvert à tous). Le premier contexte est un scénario collaboratif classique alors que le second est un cas spécial d’interaction en binôme, soit le binôme enseignant – étudiant.

Nous allons également démontrer par une expérience, comment utiliser la corrélation entre le résultat de l’apprentissage par MOOC et les traces occulaires des étudiants pour créer un outil de feedback qui permet d’améliorer leurs résultats d’apprentissage. L’outil que nous proposons fonctionne en orientant le niveau d’attention des étudiants lors du visionnage de vidéos d’un MOOC. Nous allons aussi montrer que le regard peut être utilisé comme indicateur de références verbales par l’enseignant dans les vidéos MOOC, et de ce fait, nous permettre d’améliorer le processus d’apprentissage par MOOC des étudiants.

Cette thèse comprend cinq études. La première étude se passe dans un contexte collaboratif où les partenaires doivent comprendre un programme informatique. Dans cette étude, nous examinons la corrélation entre les traces occulaires des partenaires, leur dialogue et le niveau de compréhension atteint à la fin de la tâche.

Les quatre études suivantes ont été faites dans le contexte de l’apprentissage avec les MOOCs. La première étude MOOC explore la corrélation entre la performance des étudiants et leur niveau d’attention. La seconde étude, faite avec deux eye-trackers, examine les traces du regard en individuel et en collaboration de même que leurs corrélations avec les résultats de l’apprentissage. Cette étude explore également l’idée d’activer la connaissance des étudiants, avant qu’ils ne reçoivent du matériel éducatif, et l’effet des différentes méthodes d’activation de cette connaissance sur les traces du regard des étudiants et leurs résultats d’apprentissage.

La troisième étude, durant laquelle nous avons créé un outil de feedback basé sur les deux premières expériences utilisant des MOOCs, démontre que les variables que nous avons proposé pour mesurer l’attention des étudiants peuvent être utilisées pour fournir un feedback sur leurs traces occulaires. Nous montrons également qu’utiliser cet outil de feedback permet

d'améliorer les résultats d'apprentissage des étudiants et leurs niveaux d'attention. La quatrième et dernière étude MOOC montre que l'ajout d'informations concernant le regard de l'enseignant aide les étudiants dans leur processus d'apprentissage. Lorsque l'endroit où se porte le regard de l'enseignant est affiché, la difficulté perçue du contenu diminue de manière significative par rapport aux moments où le regard n'est pas indiqué.
En résumé, par cette thèse, nous démontrons que le regard peut être utilisé pour comprendre, offrir du support et améliorer les interactions en binôme afin d'augmenter la possibilité d'atteindre un niveau de succès plus important dans la résolution de tâches.

Mots clefs : L'oculométrie, oculométrie double, Cours en ligne ouvert à tous, Analyse de l'apprentissage, la compréhension du programme, la programmation en paire, Co-résolution de problèm.

Contents

List of Figures

List of Tables

1 Introduction

"The eyes are the mirror of the soul and reflect everything that seems to be hidden."
- Paulo Coelho

1.1 Motivation

What happens during collaboration? When two persons collaborate, they try to build up a shared understanding or to learn a new topic from a particular domain. Very often, the underlying cognitive processes are hidden from the observer. In order to support collaboration, it is important to understand those underlying cognitive processes. The theme of this thesis is to approach this problem using a learning analytic approach.

Our learning analytics view of supporting collaborative work and collaborative learning is like a cybernetic control [Jermann, 2004]. In a cybernetic control, the current value of the controlled environment is compared against a reference, and if need be, some adjustments are made. In terms of collaborative work and learning, we first differentiate the behavioural patterns captured from a collaborative environment between successful and unsuccessful collaboration. Once we know the different behavioural patterns corresponding to a successful collaboration, we might be able to provide constructive suggestions in other collaborative settings. This assumption requires careful experimentation, where we try to understand the cognitive processes underlying collaboration. The major focus of these experiments is to collect behavioural data that can help us build this understanding.

In this thesis, we captured the behaviours of the collaborating partners using their gaze data and the dialogues. In the present decade, the costs and sizes of eye-trackers have decreased down to a level that the experiments are not limited to the laboratory settings. With the current advent in technology, it has been possible to collect high quality gaze data with high accuracy and precision. This data enables us to model the cognition underlying the collaborative work and learning and later to develop feedback systems that can support collaborative activities.

1.2 Research context

This thesis proposes a learning analytic approach to understand the collaborative processes. The data collection is a major part of this approach. We will use gaze data as the main behavioural measure in our experiments. As shown in figure 1.1 our work lies at the confluents of several domains:

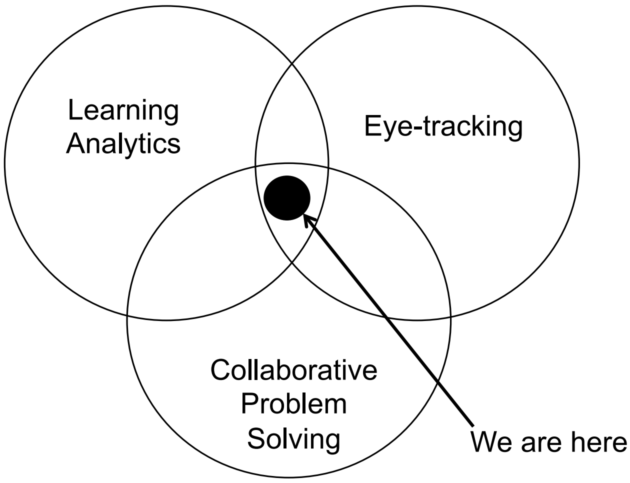


Figure 1.1 – *The placement of this dissertation work within the relevant research areas.*

1) *Collaborative problem solving*, our main problem statement comes from this domain. We study dyadic interaction. The goal of the dyadic interaction, in our case, as will be addressed in this thesis, is to build up a shared understanding. This dyadic interaction can either be with a collaborating partner (pair programming); or it can be a teacher-student pair in a Massive open online course (MOOCs), with the special case of the teacher always being the leader in the interaction.

2) *Learning analytics*, provides the approach to answer the questions raised from the two contexts: pair programming (collaborative problem solving) and MOOCs (teaching). The prime motive is to approach the dyadic interaction processes as a environment controlled via a cybernetic control. The first stage is to understand the processes and the second stage is to develop the feedback system based upon our understanding.

3) *Eye-tracking*, provides us the methodology to understand and support dyadic interaction. We will use the gaze data to explain the differences among different levels of success (level of understanding in pair program comprehension and learning outcome in MOOCs) in such interactions. Later, we will develop the gaze-aware feedback tools based on the different gaze patterns corresponding to the different collaborative outcomes.

1.3 Global research questions

Through this dissertation, we try to answer the following research questions:

How are gaze patterns, dialogues and shared understanding related to each other in a collaborative setting? We hypothesise a complex (triangular) relationship between these three constructs. The intertwining of gaze patterns and dialogues leads to a given level of shared understanding; and we are interested in finding the gaze patterns and dialogues that can differentiate among different levels of collaborative success.

How can we measure the attention of a student in a Massive Open Online Course (MOOC) lecture specially considering the teacher-student dyad as a social unit of interaction? Moreover, how are these attention measures related to the student performance? Considering the teacher-student dyad as our social unit of interaction, lets us control a post-hoc variable while analysing the collaboration. This variable is the different roles (leader, follower) acquired by the different participants in pairs. The goal of collaboration (mainly for the student) is to build up a shared understanding of the topic at hand. By capturing the attention automatically, we might be able to develop a gaze based "cybernetic control system" as described in Section 1.1.

How can we improve the learning experience in terms of attention in MOOC lectures? Provided, we are able to find differences in gaze patterns of successful and unsuccessful students, the last question we will try to answer is "how can we improve the learning experience in MOOC lectures". One way to provide a high quality learning experience is to provide the feedback based on the answer of the previous question. In other words, we want to evaluate the effectiveness of the attention measure(s) we will develop in this thesis.

1.4 Thesis Roadmap

This dissertation is organised as follows:

In the next chapter, we will provide a brief overview of the use of eye-tracking to distinguish different levels of task based performance, task difficulty, problem solving strategies, and expertise; both in the individual and collaborative settings. Moreover, we will also give a brief overview of previous work done to understand cognitive mechanisms underlying collaborative problem solving and learning.

Chapter 3 will present a dual eye-tracking study to find the relationship between gaze patterns, dialogues and the level of understanding. The study is contextualised within a pair programming setting, where pairs of programmers collaborate to understand a given program.

In chapter 4, we will present the design and results from an exploratory eye-tracking study in the MOOC context. The data collected from this study was used to develop measures to capture the students' attention in a MOOC and to find the relation between these measures and the students' performance.

Chapter 5 will present the design and results from the second (dual) eye-tracking study in the MOOC context. We built upon the results from the study in chapter 4 and also introduced some ecologically valid changes in the study presented in chapter 4.

In chapter 6, we will present the design of a gaze-aware feedback system to support MOOC students while watching the video lectures. We will also report on a study, where we compared this gaze-aware feedback system against the absence of any feedback.

Chapter 7 will present a study where we recorded the eye-tracking data from a MOOC teacher, while he was recording his lecture. We describe the effect of showing the teacher's gaze on the MOOC lecture on the navigation patterns of students.

Finally we will conclude with a summary and general discussion about the contributions of this work. This chapter will also explain the limitations and the implication of our work for future research.

2 Related Work

"Learning analytics is the use of intelligent data, learner-produced data, and analysis models to discover information and social connections, and to predict and advise on learning."
- George Siemens [a]

[a]Source: "http://www.elearnspace.org/blog/2010/08/25/what-are-learning-analytics/"

2.1 Introduction

Eye-tracking provides researchers with an unprecedented access to the users' attention. The eye-tracking data is rich in terms of the temporal resolution. With the advent in eye-tracking technology, the apparatus has become compact and easy to use without sacrificing much of the ecological validity during the controlled experiments. Previous research had shown that eye-tracking can be useful to unveil the cognition that underlies the interaction between collaborating partners, the different strategies that experts chose to solve problems at hand. Eye-tracking was also shown to be useful to differentiate the strategies which led to success from those which could not. Gaze has also shown to be related to dialogues among collaborating partners.

In this chapter, we will present examples from the previous research showing the usefulness of eye-tracking in learning analytics. We start with reviewing research carried out using gaze as an analytics tool, where we show how different studies used eye-tracking data: *1)* to find the key moments in an interaction; and *2)* to find the expert strategies for problem solving. We will then review two exemplar fields where eye-tracking had been used as an analytic tool: program comprehension and online learning. Then we will present examples of studies using eye-tracking data to quantify cognition at different temporal granularities.

We will not present an exhaustive literature review on the previous research done in the field of eye-tracking. Instead, we chose studies that exemplified major topics in the eye-tracking research conducted in major problem solving fields, for example, the insight problem solving (matchstick arithmetic), games and sports (boxing and chess), and the procedural problem solving (arithmetic word problems and program comprehension).

As we said earlier that another closely related source of data for analytics are dialogues. We will show what relations have previous studies found in gaze data and dialogues (or explicit references) during interactions and problem solving. Moreover, we review the studies carried out using dialogues as a source of analytics data to find different problem solving strategies across different expertise levels or across different performance levels.

2.2 Gaze as an analytics tool

Gaze had been found to be closely related to different strategies, expertise, task-based performance and dialogues. In this section, we review past research using gaze data to identify different strategies across the different expertise and performance levels. We also review the studies establishing the relation between the dialogues and the gaze.

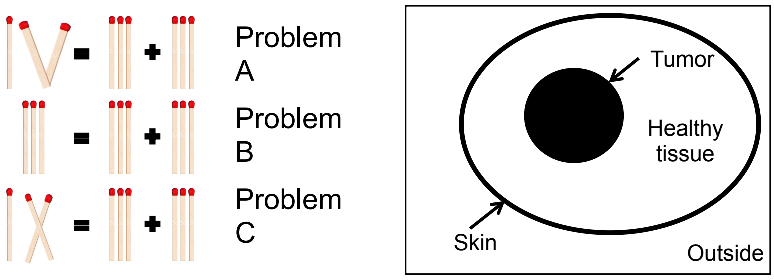
2.2.1 Gaze and problem solving

Eye-tracking had been used in numerous studies to find the relation between task-based performance or expertise with the gaze-patterns. In this section, we report a few exemplar experiments.

Knoblich et al. [2001] used eye-tracking to study how participants solved insight problems. As an example of insight problems Knoblich et al. [2001] used *matchstick arithmetic problems* (Figure 2.1a). In a typical matchstick arithmetic problem, the participant is asked to correct an incorrect arithmetic equation. The equation uses the Roman numerals. The participant had to move one and only one matchstick from one position to other, in order to correct the equation. In figure 2.1a, problem A was solved by changing “IV” to “VI”; problem B was solved by changing “+” to “=”; and problem C was solved by changing “IX” to “VI”. Problems B and C were more difficult than problem A, because solving problem B involved changing one of the operators and solving problem C involved changing the partial structure of a numeral.

The major difficulty in the insight problems is occurrence of impasses due to two different reasons. In contrast to usual problem solving where the problems are resolved gradually, the insight problems are solved suddenly [Thevenot and Oakhill, 2008]. The two reasons for impasses are based on this fact only. *1)* Usual problem solving involves minimising the distance between the problem state and the solution state. In insight problems, impasses occur when the participant finds that his/her actions do not reduce this distance [MacGregor et al., 2001]. This is also known as *progress monitoring theory*; and *2)* impasses could also occur if the participant starts from an incorrect initial representation of the problem [Knoblich et al., 1999]. This is also known as *representational change theory.*.

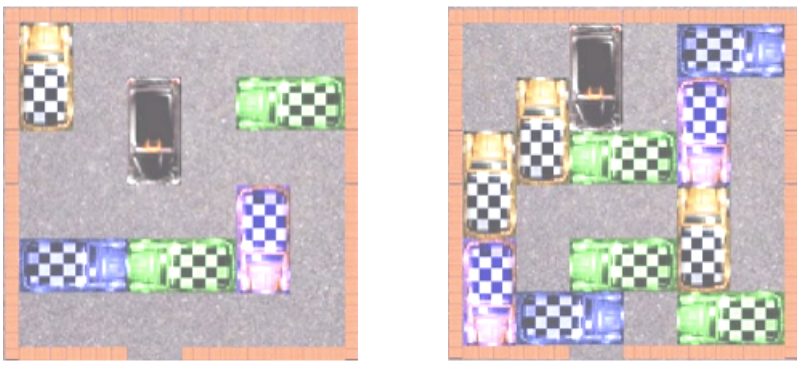
Knoblich et al. [2001] measured the fixation time on different chunks (each Roman numeral) of matchsticks in each of the problems. The results showed that during an impasse for difficult problems (B and C) participants were simply staring at the problem, i.e., they had fewer and longer fixations. Also in the later phases of successfully solved problems Knoblich et al. [2001] found more fixations on the result side of the equations. For example, during successful solutions to problem C, the participants looked more at the "X" part of "IX", thus showing the more emphasis on the key part of the result side.



(a) (b)

Figure 2.1 – *(a) Examples of matchstick arithmetic problems used by Knoblich et al. [2001]. The problem "A" is an easy problem, and problems "B" and "C" are the difficult ones. (b) A typical example stimulus for the "Duncker's radiation problem".*

Jones [2003] used another example of insight problem called *Car Park* problem (figure 2.2) to find the relation between the problem solving processes and the gaze data. The goal of the car park problem is to manoeuvre a car out of a parking space. The parking space has other cars as well, which can be moved only in their initial orientation. The authors looked at the fixation time three moves prior to the object car move and three moves after the object car move. The fixation time on the problem was longer for the object car move, than that in the prior or succeeding moves to the object car move. Moreover, non-solvers spent significantly more time on the free area than the solvers.



(a) (b)

Figure 2.2 – *Car Park problem used by Jones [2003]. The object car is coloured in black.*

Grant and Spivey [2003] used another example of insight problem called the *Duncker's radiation problem* (figure 2.1b), which is defined as follows:
"Given a human being with an inoperable stomach tumour, and lasers which destroy organic tissue at sufficient intensity, how can one cure the person with these lasers and, at the same time, avoid harming the healthy tissue that surrounds the tumour?" - Grant and Spivey [2003].

Grant and Spivey [2003] measured the fixations on the "skin", "tumour", "inside" and "outside" (Figure 2.1b). The results showed that there was significantly more time spent on the skin during successful solution, than that during unsuccessful solutions. This showed that the skin is a critical feature in problem solving process. This led the authors to conduct another experiment where they compared highlighting the "skin" (critical feature) versus highlighting the "tumour" (non-critical feature). The results from the second experiment showed that highlighting the critical feature led to significantly more correct solutions than the condition with highlighting the non-critical feature.

Thomas and Lleras [2007] also used the Duncker's radiation problem to establish the relation between the problem solving processes and the gaze data. The authors manipulated the eye-movements of the participants in four different ways as shown in figure 2.3: *1) embodied–solution*, where participants' saccades crossed the skin many times; *2) areas-of-interest*, where the participants had the same patterns as the previous group but they had shorter saccades; *3) repeated-skin-crossing*, where participants crossed the skin between the same two points only; and *4) tumour-fixation*,where participants looked only at the tumour.

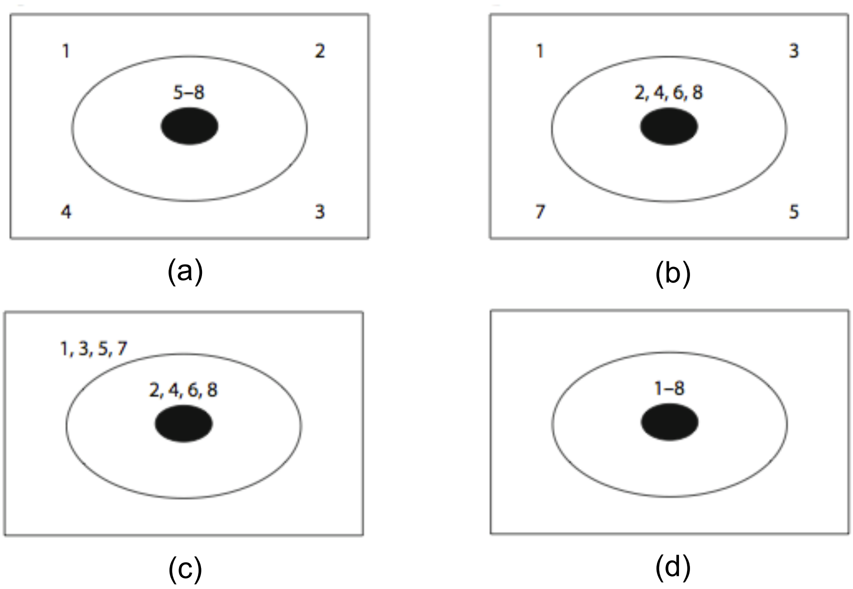


Figure 2.3 – *Typical example of tumour task used by Grant and Spivey [2003] and Thomas and Lleras [2007]. In the case of Thomas and Lleras [2007], the authors forced the participants to look in certain way (the numbers represent the order of objects to be looked at); (a) shows the embodied-solution group; (b) shows the areas-of-interest group; (c) shows the repeated-skin-crossing group; and (d) shows the tumour-fixation group.*

The results from Thomas and Lleras [2007] showed that by forcing the participants to look only in a specific way the success rate of the solution can actually be manipulated. For example the the success rate was found to be increasing in the following order: *1)* repeated-skin-crossing, *2)* tumour-fixation, *3)* areas-of-interest, and *4)* embodied-solution. The two studies, about the Duncker's radiation problem, showed that given the correct feedback/intervention, the task-based success could be improved.

Just and Carpenter [1976] used eye-tracking to explain different cognitive processes underlying the problem solving in a *mental rotation task*. The participants had to perform a *same-different* task for three angles of rotations (figure 2.4). For the participants, there were three main components of the task: first, to figure out what parts were to be rotated; second, how much the parts had to be rotated; and third, whether after rotation the two figures were the same or not. The authors called these three components as search, transformation and comparison, and confirmation.

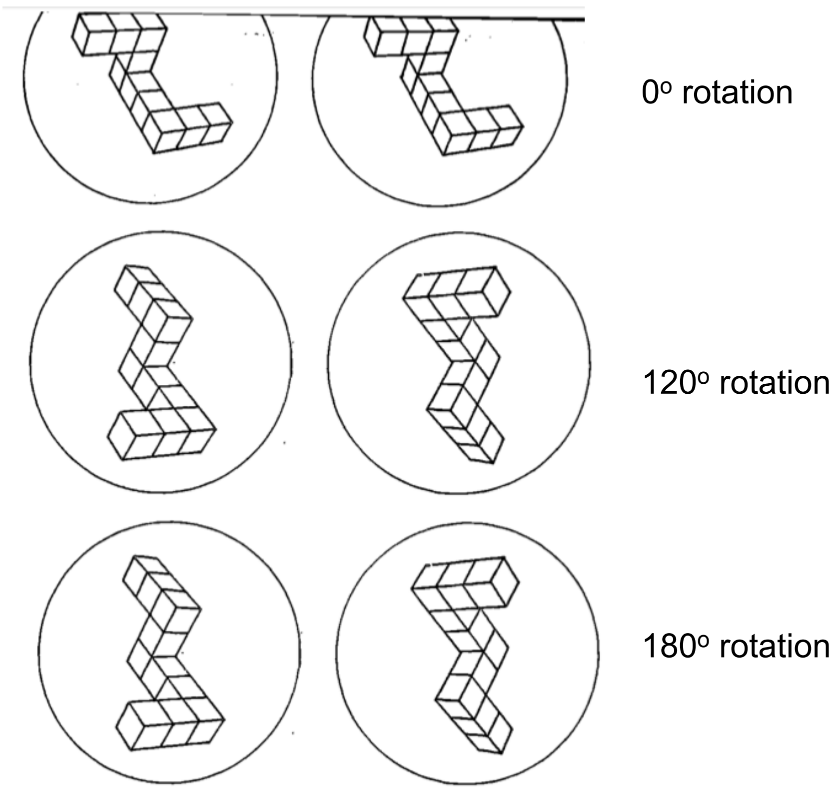


Figure 2.4 – *Different rotation angles used by Just and Carpenter [1976].*

Overall results from Just and Carpenter [1976] show that there was a common pattern across the three rotation types (0, 120 and 180 degrees). The participants switched between the figures three times (left-right-left-right). The number of such switches increased with the increase in the rotation angle. Further, the authors divided the fixations into three categories: *1)* fixations at the center, *2)* fixations at the arm with the third face of the cube visible (open), and *3)* fixations at the arm with the third face of the cube not visible (close). The authors constructed the scan paths from these categorised fixations; and further categorised the scan paths to represent the three components of the problem solving process. The results showed that the time intervals for the three processes were different and they increased with the increase in the rotation angle.

Ripoll et al. [1995] used eye-tracking data to analyse the different visual search activities of the boxers across the different levels of expertise (expert, novice and intermediate) and task complexity in two different experiments. The participants had to solve *French boxing* [1]situations. The opponent (virtual) was filmed and projected on the screen. The participants had to respond using a joystick. Each participant was asked to respond to five situations: left and right attacks, left and right feints and the openings. The authors divided the fixations onto different body parts like: head, trunk, arms/fists, pelvis and legs. The results showed that the experts made significantly more fixations on head than the novices and intermediates; while they had no fixations on the lower body parts. The authors suggested that the information about the lower parts might had come from the peripheral for the experts. Moreover, the novices focused more on the arms/fists than the experts and the intermediates; while the intermediates focused more on the trunk than the novices and experts.

Abernethy and Russell [1987] used racquet sports to explore the relationship between the gaze patterns and the different levels of expertise (experts and novices). The participants were all badminton players. The stimulus for the gaze experiment was prepared in a similar manner as in Ripoll et al. [1995]. The only difference between the two experiments was that some of the frames in the stimulus used by Abernethy and Russell [1987] were occluded. The occlusions were deliberately placed either at the body of the player or at the entire frame prior/after the racquet-shuttle contact. The experimental task was to predict the landing position of the shuttle. The analysis was carried out by categorising the fixation into five categories: racquet/arm, shuttle, trunk, head, legs/feet. The results showed that the experts focused more on the racquet and arm of the opponent; while novices focused more on the head and the trunk of the opponent. These results were the opposite of the results found by Ripoll et al. [1995]; this shows the sensitivity of the gaze patterns towards the task specificities.

Kaller et al. [2009] compared the gaze patterns of participants across the different task difficulty levels during a visuospatial task of *Towers of London* (Figure 2.5). The order for the presentation of the start and the goal was a between subject variable used by [Kaller et al., 2009]. Half of the participant saw the problem with start on the left (as shown in the figure 2.5, SG group). The other half saw the opposite representation (figure2.5, GS group). The authors did not find any differences in terms of performance across the two experimental groups. However, the participants initially (first 144 observations per participants) looked more at the left diagram more than the right diagram irrespective of the state (start or goal) it was displaying. Considering the gaze shifts between left and right sides during initial thinking time (time between the presentation of the problem and onset of the first action), the authors found that the gaze shifts were highly influenced by the fact whether the participant first looked at the

[1]French boxing, also known as French kickboxing or French foot fighting, for details, see here

goal or start state. There were more gaze shifts among the states when the participants started from the goal state than those when the participants started from the start state.Moreover, there was a high amount of gaze directed towards the start state during the initial phase of the solution execution phase (when participants started moving the pegs) across both the SG and GS groups. This duration increased with the increase in task difficulty. The authors concluded that there is a strong dependency between the personal preferences and the gaze patterns; and between the task difficulty and the gaze patterns.

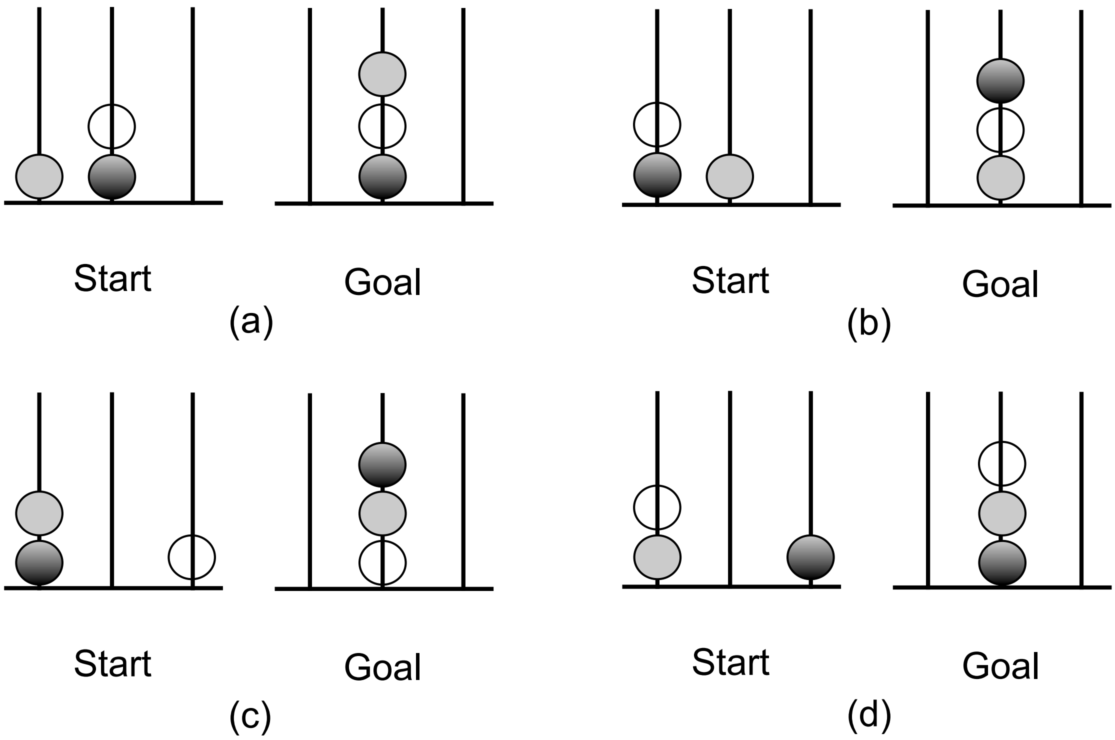


Figure 2.5 – *Tasks used by [Kaller et al., 2009]. (a) Type 1: one-move problem. (b) Type 2: two-move problem. (c) Type 2: three-move problem, without intermediate step. (d) Type 4: three-move problem, with intermediate step.*

Hegarty et al. [1992] used the gaze data to understand how students solve the *arithmetic word problems*. To solve the problem shown in figure 2.6, the students had to make the relation using the second sentence as "Price at ARCO = Price at Chevron + 5 cents". The authors used four versions of the same word problem using consistent and inconsistent language (using "this" instead of the shop name); and using two different relational words ("more" and "less"). The authors give two major problems faced by the students in solving inconsistent problems: *1)* using "less" as relational inverses the actual relation; *2)* students make mistake in assigning the noun to "this". The authors divided the students using their accuracy (high and low-accuracy) in solving the arithmetic word problems to concentrate more on the high-accuracy students and their gaze patterns. The authors found rereading patterns, for high-accuracy students, were in a way that every rereading iteration had progressively smaller chunks of text on any given line. Moreover, for every rereading iteration, these students focused on numbers more than the other information. Also, they reread the variable names and the relational terms in inconsistent problems than in the consistent problems.

	Consistent language	**Inconsistent language**
Relational term "MORE"	1. At ARCO gas sells for $ 1.13 per gallon. 2. Gas at Chevron is 5 cents more per gallon than gas at ARCO. 3. If you want to buy 5 gallons of gas, 4. how much will you pay at Chevron?	1. At ARCO gas sells for $ 1.13 per gallon. 2. This is 5 cents more per gallon than gas at Chevron. 3. If you want to buy 5 gallons of gas, 4. how much will you pay at Chevron?
Relational term "LESS"	1. At ARCO gas sells for $ 1.13 per gallon. 2. Gas at Chevron is 5 cents less per gallon than gas at ARCO. 3. If you want to buy 5 gallons of gas, 4. how much will you pay at Chevron?	1. At ARCO gas sells for $ 1.13 per gallon. 2. This is 5 cents less per gallon than gas at Chevron. 3. If you want to buy 5 gallons of gas, 4. how much will you pay at Chevron?

Figure 2.6 – *Arithmetic word problems used by Hegarty et al. [1992]. There were four versions with consistent and inconsistent language and with relational words "more" and "less".*

Ballard et al. [1992] used an eye-tracking to study the hand-eye co-ordination during sequential tasks, such as copying a model. The participants were asked to copy a model using the blocks provided in a separate area on the screen. The participants had to copy a given model in terms of both the colour of the block and its position relative to the other blocks. The task complexity was determined by the number of blocks involved in the model. The authors found that there was a clear cognitive algorithm to complete such tasks: *1)* participants looked at a block in the model and remembered its colour; *2)* they looked at the same colour block in the source area; *3)* they picked up that block; *4)* they revisited the block in the model and remembered its position; *5)* they moved the block from the source area to the copying area. The authors observed that the fixations on the blocks were either at the onset of the hand movement or at the end of the movement.

Charness et al. [2001] conducted a study to compare the gaze patterns of expert and intermediate chess players. The participants were asked to make the best move for a given chess position as quickly and as accurately as possible. The experts were faster and more accurate than the intermediate players in terms of making the move. The authors observed that the experts looked more at the vacant blocks than the intermediate players; and while fixating on the pieces the experts spent more time than intermediate players on the relevant pieces. Also experts made longer saccades than the intermediate players. Charness et al. [2001] concluded that the experts encoded the configurations more than the individual pieces; while the intermediate players encoded the positions of individual pieces.

Reingold et al. [2001] used the gaze data of expert chess players to find out how they encoded a given chess position. The authors conducted a study with different levels of chess players (novices, intermediates and experts) and two tasks. In the first task, participants were shown two kinds of chess configurations (figure 2.7): random and original game configurations. Each configuration had a modified form as well where the authors modified one of the pieces in the gaze contingent zone, i.e., the zone that was clearly seen by the participants; rest of the visual stimuli was blurred (the bright circular zones in each of the configurations in figure 2.7).

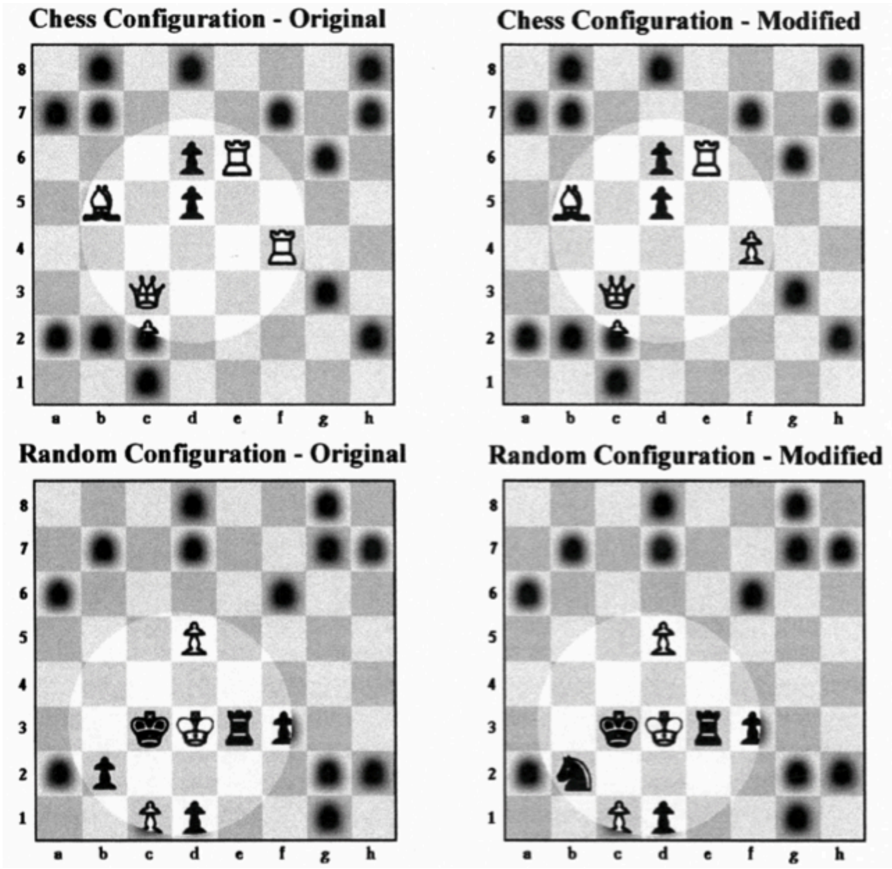


Figure 2.7 – *Chess positions used by Reingold et al. [2001].*

Participants were asked to detect the modified piece. In the second task, the participants had to detect whether there was a *check situation* on a 3 X 3 chess board. For the first task, the authors calculated the area of visual span as the number of squares looked at by the participant. The results showed that, in the original game configurations the experts were faster to detect the modification and had a larger area of visual span, than those in random configurations. Reingold et al. [2001] found no differences for the novices and intermediate players across the two configurations. In the check detection task, experts made fewer fixations on pieces than the less-skilled players. The authors concluded that the experts encode a larger chunk of the configuration than the novices as they use their foveal and parafoveal regions to get inter piece information as suggested previously by Chase and Simon [1973].

Harbluk et al. [2007] used the car drivers' gaze data to understand how their "on-road" cognition worked. The drivers were asked to complete three 4-km drives with additional cognitive task of arithmetic addition: easy, with one digits addition (6 + 3 = 9) and difficult, with two digits addition (46 + 37 = 83), and no task. The drivers looked more on the forward view in task conditions than in no task condition. However, they paid less attention to the mirrors, instruments and the peripherals during task conditions than in no task conditions. The level of difficulty in the cognitive tasks elevated these differences. Also the subjective ratings about the cognitive load, reduction of safety and distraction was found to be increased from no task to easy task to difficult task conditions.

The following table summarises the findings reported in previous studies:

Table 2.1 – *Different factors in problem solving and their gaze correlates. Rows marked with "*" represent the studies where an intervention/feedback was introduced, that resulted in a significant improvement in task-based success.*

Paper	Task	Discriminating factor
Knoblich et al. [2001]	Matchstick arithmetic	Task difficulty and success
Jones [2003]	Car parking	Task difficulty problem solving strategy and success
Grant and Spivey [2003] *	Duncker's radiation (tumour task)	Task difficulty and success
Thomas and Lleras [2007] *	Duncker's radiation (tumour task)	Success
Just and Carpenter [1976]	Mental rotation	Task difficulty and problem solving strategy
Ripoll et al. [1995]	Boxing	Expertise
Abernethy and Russell [1987]	Racquet sports	Expertise
Kaller et al. [2009]	Towers of London	Task difficulty
Hegarty et al. [1992]	Arithmetic word problems	Problem solving strategy
Ballard et al. [1992]	Copying a model	Problem solving strategy
Charness et al. [2001]	Chess	Expertise
Reingold et al. [2001]	Chess	Expertise
Harbluk et al. [2007]	Driving a car	Task difficulty

2.2.2 Gaze in communication and referencing

Gaze and speech are coupled. Previous studies had shown a strong relation between dialogues and/or speech of the speaker and his/her gaze. Also there were studies showing the relation between the speakers' dialogues and listeners' gaze. In this section, we review some of the studies which shed some light on the gaze-speech coupling.

Meyer et al. [1998] showed that the time duration between looking at an object and naming it is between 430 and 510 milliseconds. In their experiment, the participants were shown line diagrams of a few objects and were asked to name them. Griffin and Bock [2000] showed that there exists an eye-voice span of about 900 milliseconds. The eye-voice span denotes the time between looking at a picture and start to provide a short explanation to it. Zelinsky and Murphy [2000] had shown that there was a correlation between the time spent gazing at an object and the spoken duration for naming that object. In the experiment conducted by Zelinsky and Murphy [2000], the participants were shown objects with one (cat, car) and two (aircraft, basket) syllable names. The authors found that the participants looked at two syllable objects for longer durations than they looked at one syllable object.

Allopenna et al. [1998] conducted an experiment to measure the time duration between the speaker's verbal reference to an object and the listeners' gaze-onset on the referred object. The authors used stimulus images as shown in the figure 2.8. The main function for the *referent* and the *cohort* (figure 2.8) was to provide the same audio cue to the listener. For example, both the words "beaker" and "beetle" would activate the same an initial tendency to look at the object in the image. This introduced a situation where the listener had to pay attention to the whole word. Allopenna et al. [1998] showed that the mean delay between hearing a verbal reference and looking at the object of reference (the listeners' voice-eye span) was between 500 and 1000 ms.

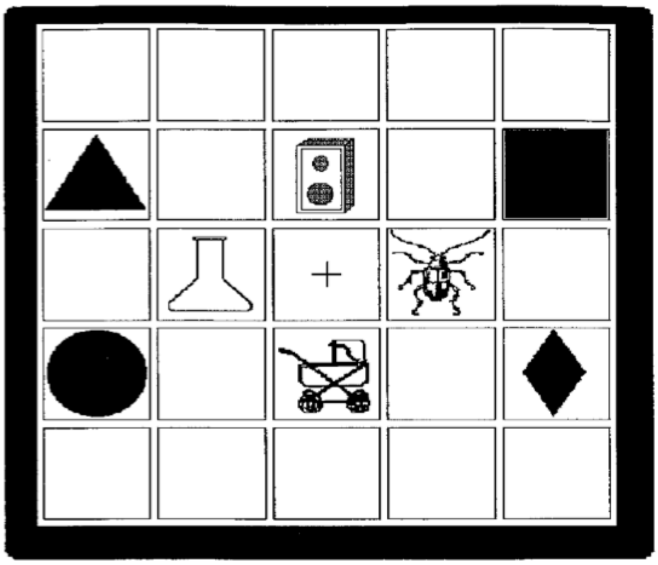


Figure 2.8 – *Stimulus image used by Allopenna et al. [1998]. In this particular image the the beaker is the referent, beetle is the cohort, speaker is speaker and carriage is unrelated.*

Richardson et al. [2007] proposed the eye-eye span as the difference between the time when the speakers started looking at the referent and the time when listeners looked at the referred object. In a dual eye-tracking experiment, Richardson et al. [2007] asked one of the participants in each pair to narrate the relationship between the characters in the famous TV series "Friends" to the other participant in the pair. The authors measured the time lag between the speakers looking and referring at a specific actor and the listeners looking at the same actor. This time lag was termed as the cross-recurrence between the participants. The results show that the cross recurrence was correlated with the correctness of the answers given by the listeners in a comprehension quiz. The average cross-recurrence was found to be between 1200 and 1400 milliseconds. This time was consistent with the additions of eye-voice span found by Griffin and Bock [2000] and voice eye-span found by Allopenna et al. [1998].

Jermann and Nüssli [2012] extended the concept of cross-recurrence in a pair programming task, by enabling the remote collaborators to share their selections on the screen. The authors found the similar levels of cross-recurrence as it was found by Richardson et al. [2007]. The participants in this dual eye-tracking experiment were asked to collaboratively understand a JAVA program of about 200 lines of code. The selections made by one participant in each pair were also shown to the other participant in the pair. Jermann and Nüssli [2012] found that the cross-recurrence levels were higher when there was a selection present on the screen than the times when there was no selections on the screen. Moreover, the cross-recurrence was higher, in the case, when a selection was followed by a verbal explanation.

Gergle and Clark [2011] conducted a dual eye-tracking study where the participants completed a collaborative reference elicitation task. The participants were given four replicas for the same sculpture. The key task for the participants was to find the correct replica. To find the correct replica the participants were required to discuss amongst themselves the different objects in each replica and matching them with the original sculpture. There were three conditions in the experiment: *1)* the pair was seated side-by-side, *2)* the pair was seated across the table, and *3)* the pair was allowed to move. The authors found that the mobile pairs produced more local references (including pronouns like “this”, “here”) while the seated pairs produces more elongated references (with additional modifiers). Moreover, the authors also found that the gaze overlap between the partners was lowest when the references were local as compared to when the references had location modifiers.

Duchowski et al. [2004] compared three modalities of assisting a referrer’s deictic references to his partner in a virtual collaborative environment. The three assisting cues were: head rotation, head and eye rotation, head and eye rotation with the light-spot over the target. The participants were asked to verbally identify the target selected by the referrer. The authors concluded that the reference disambiguation is fastest when the light-spot was shown along with the head and eye rotations.

Cherubini and Dillenbourg [2007] explored the relation between the ability to explicitly refer at something in a collaborative map annotation task, and the success in the task. The participants were asked to plan a music festival around the university campus by annotating a map with parking spots, places for drinks and stages. The participants were given a chat tool. The chat application had two modalities. In one of the modes the participants could link the the places they were talking about in the map with what they wrote in the chat; while in the other mode there was no such facility. The results showed that the with the explicit referencing enabled the pairs were faster in completing the task; and they had more concrete references in the terms of message length, compared to the modality without the facility of explicit referencing.

2.2.3 Gaze and program understanding

Several studies have been conducted to show the different aspects of the relation between gaze and task performance in the context of programming. The studies can be classified based on the granularity of eye-tracking analysis and based on the type of study. Concerning the eye-tracking setup, most of the analyses conducted so far rely on a partition of the screen into large Areas of Interest (AOIs). The screen is typically divided into regions that correspond to elements of the interface (e.g. a panel for code, the console, and a panel for diagrams); and the analyses were usually focused on the proportion of time spent, and the transitions between the areas of interest; which are then related to task-based performance.

Pietinen et al. [2010] gave a new metric, to measure joint visual attention in a co-located pair programming setup, using the number of overlapping fixations and use the fixation duration of overlapping fixation for assessing the quality of collaboration. In another study Pietinen et al. [2008] presented a possible design of the eye-tracking setup for co-located pair-programming and addressed some of the problems regarding setup, calibration, data collection, validity and analysis. Bednarik and Tukiainen [2006] examined coordination of different program representations in a program understanding task. Experts concentrated more on the source code rather than looking at the other representations. The different representations were taken to be different AOIs. Bednarik et al. [2006] tried to relate the information types (by Good and Brna [2004]) to the gaze among the four AOIs (Code, Output, Control Panel and Animation of program). The authors concluded that presence of information type (e.g. high-level or low-level) in the comprehension summary does not correlate to the fact that that the target program was correctly comprehended.

Romero et al. [2002] compared the use of different program representation modalities (propositional and diagrammatic) in a expert novice debugging study where experts had a balanced shift of focus among the different modalities than that for the novices. Sharif et al. [2012] emphasised the importance on code scan time in a debugging task and conclude that experts perform better and have shorter code scan time. Hejmady and Narayanan [2012] compared the gaze shift between different AOIs in a debugging IDE. The authors concluded that good debuggers were switching between code and the expression evaluation and the variable window rather than code and control structure and the data structure window.

2.2.4 Gaze and online/multimedia learning

Use of eye tracking in online education has provided researchers with insights about students' learning processes and outcomes. Van Gog et al. [2005b] used eye tracking data to differentiate the expertise levels in the different phases of an electrical circuit troubleshooting problem and conclude that experts focused more on the problematic area than the novices. Van Gog et al. [2005a] used eye tracking data to provide feedback to students about their actions while troubleshooting an electrical circuit and found that the feedback improved the learning outcomes. Van Gog et al. [2009] found that displaying an expert's gaze during problem solving guided the novices to invest more mental effort than when there was no gaze displayed.

Amadieu et al. [2009] used eye tracking data to find the effect of expertise, in a collaborative concept-map task, on the cognitive load. The authors divided the concept-map structure into two categories: hierarchy based and network based. The authors concluded that the average fixation duration was lower for the experts (when they produced the hierarchy based concept-map) indicating less cognitive load on experts than novices. In an experiment, where the participants had to learn a game, Alkan and Cagiltay [2007] found that the good learners focused more on the contraption areas (areas that appeared strange or unnecessarily complicated) of the game while they think about the possible solutions. Slykhuis et al. [2005] found that students spent more time on the complementary pictures in a presentation, than on a decorative picture.

Mayer [2010] summarised the major results of research on eye tracking in online learning with graphics and concluded that there was a strong relation between fixation durations and learning outcomes; and visual signal guided students' visual attention. In another study to compare the effect of colour coded learning material, Ozcelik et al. [2009] found that the learning gain and the average fixation duration were higher for, and hence more mental effort was put by, the students who received colour coded material than those who received non-colour coded material.

2.3 Dual eye-tracking and collaborative problem solving

Two synchronous eye-trackers can be used for studying the gaze of two persons interacting to solve a problem. It gives a chance to understand the underlying cognition and social dynamics when people collaborate to solve problems at hand [Nüssli, 2011]. In a collaborative task of findings bugs in a program, Stein and Brennan [2004] showed that the pairs who had their gaze displayed to their partners took less time in finding the bugs than those pairs who had no information about their partners' gaze.

Sangin et al. [2008] used a knowledge awareness tool (KAT) to inform the pair about their partners' knowledge about a certain topic in a collaborative concept map task. The participants were asked to answer a pretest before the actual collaborative task. From participants' responses, the authors built a knowledge awareness tool and displayed it to their partners while they collaborated on a concept map. From the gaze data analysis, the authors found that the participants looked at the KAT the most in the beginning of the collaboration, in order to have an assessment about their partners' knowledge. There was also a positive correlation between the gaze on the KAT and participants' relative learning gain. The authors found that the participants looked more at the KAT when the partners' provided a verbal cue about their knowledge or when they provided a new information.

From the same collaborative concept map experiment as Sangin et al. [2008], Liu et al. [2009] found that the gaze data of the pair is predictive of the expertise in the collaboration. The authors framed the whole interaction as a sequence of concepts looked at. The authors then use Hidden Markov Models to predict the outcome of posttest and achieved an accuracy of 96.3%.

Nüssli et al. [2009] used dual eye-tracking data to predict success in Raven [2] progressive matrices and Bongard problems [3]. The authors used a collaborative versions of the problems, where they partitioned the problem images in a way such that the pair had to collaborate to get the correct answer. The results show that, using the gaze density and dispersion for each of the image cell, the task success could be predicted with 78% accuracy.

Jermann et al. [2010] conducted a dual eye-tracking experiment with a collaborative version of Tetris [4]. There were two *Tetriminos* falling from top of the screen which could be controlled by the two participants in the pair. The authors used social and gaze variables to predict the pair composition (expert pair, novice pair or mixed pairs). The social variables were how many times there was a conflict of interest on the stack on the bottom of the screen and how many times the players had to cross each other. The gaze variables were the proportion of gaze on the self piece, other's piece and on the stack at the bottom. The results showed that, using these variables, the pair composition could be predicted with and accuracy of 75.28%.

The following table summarises the main predictable in this section:

Table 2.2 – *Gaze as a predicting variable for success and expertise in collaborative tasks*

Paper	Collaborative task	Predictable	Predicting feature
Stein and Brennan [2004]	Program debugging	Success	Partners' gaze information
Sangin et al. [2008]	Concept map	Learning gain	Gaze on KAT
Liu et al. [2009]	Concept map	Expertise	Sequence of concepts looked at
Nüssli et al. [2009]	Raven Bongard puzzels	Success	Gaze distribution
Jermann et al. [2010]	Tetris	Pair composition	Gaze distribution and game's social context

2.4 Different levels of analytics using gaze

Time scales had been used to describe behaviour at various levels. Eye-trackers allow us to capture attention at a time scale that has more information content than the other measures like interface event logs, dialogues or gestures. In a controlled experiment, Lord and Levy [1994] found that, the duration of eye-fixations have duration of the order of 100 milliseconds,

[2]Source: "http://en.wikipedia.org/wiki/Raven's-Progressive-Matrices"
[3]Source: "http://en.wikipedia.org/wiki/Bongard-problem"
[4]Source: "http://en.wikipedia.org/wiki/Tetris"

which gives them a place at the lower end of cognitive behavioural band [Newell, 1994]. Cognitive behavioural bands have complex actions (e.g., reading or gestures) at the higher end. Anderson [2002] identifies cognitive modelling as bridging across the behavioural bands by taking the lower level bands into account. We will reuse the levels by Anderson [2002] to refer to the Task (where we usually measured understanding), Unit task (where we usually categorised dialogues) and Operations (where we usually collected raw data). The application of intertwining the gaze and dialogues will be presented in chapter 3.

2.4.1 Social granularity

With regards to the social unit of analysis, gaze had traditionally been used to assess individual cognition (e.g. eye-tracking studies about reading, program comprehension, etc.). However, in the context of dyadic interaction, a methodology was needed to describe collaborative gaze. Various measures of "gaze togetherness" had been used to indicate the quality of collaboration in dyadic interaction. In general, good collaboration features convergent gaze. Gaze togetherness increases significantly especially during verbal and deictic references. These measures of togetherness were, however, related to a global time scale; and did not consider the evolution of gaze focus during interaction.

There were different gaze-based measures of collaboration given by Richardson & Dale (2005), Cherubini et. al. (2008) and Pietinen et. al. (2010). Richardson & Dale (2005) used “gaze togetherness” as a notion of gaze cross recurrence (how much the participants are looking at the same object at the same time). Cherubini et. al. (2008) used eye tracking in a remote collaborative problem solving setup to detect the misunderstanding (distance between the referrers’ and the partners’ gaze points) between the collaborating (through chat) partners. Pietinen et. al. (2010) gave a new metric, to measure joint visual attention in a co-located pair programming setup, using the number of overlapping fixations and use the fixation duration of overlapping fixation for assessing the quality of collaboration. The problem with these measures was that they characterise togetherness on a global level or on an arbitrarily defined timespan (one could partition the interaction into “n” parts but these would not reflect the underlying interactive dynamics).

2.4.2 Temporal granularity

With regards to the temporal granularity of analyses, studies have emphasised on overall measures of individual attention. For example, studies (Romero et.al, 2002; Bednarik & Tukiainen, 2006; Bednarik et. al., 2006; Sharif & Maletic, 2010; Hejmady & Narayanan, 2012; Pietinen et. Al., 2008; Pietinen et. Al., 2010; Bednarik & Shipolov, 2011) have reported the proportion of time that subjects spent fixating on different parts of the interface. These measures indicated overall gaze behaviour (and may be correlated with expertise), but they could not serve as real-time indicators of collaboration which could be used to provide immediate feedback. In the context of dyadic interaction, the dynamics of interaction and dialogue are important indicators for collaborative knowledge building (e.g. Stahl, 2000). New gaze indicators are needed to reflect the knowledge building at the micro level.

At the level of operations, there were studies about gaze and speech coupling [Meyer et al., 1998, Griffin and Bock, 2000, Zelinsky and Murphy, 2000]. There were different notions of eye-voice span given in different studies, but all the notions point towards a strong coupling between speaker's gaze and speech. Allopenna et al. [1998] showed that the mean delay between hearing a verbal reference and looking at the object of reference (the listeners' voice-eye span) was between 500 and 1000 milliseconds. The combination of eye-voice and voice-eye coupling was that the gaze of speakers and listeners were coupled with a lag of about 2000 milliseconds. This short term coupling between speaker and listener was at the operation level only and did not inform about the relationship of gaze and dialogue in longer episodes. This is problematic when one is interested in knowledge building episodes that usually last for several utterances.

2.5 Discussion

We saw that gaze patterns correlate about the expertise, task success, task-specific strategies and deixis. In this thesis we will present new methods to analyse gaze along with the dialogues at different temporal scales. We will also show how the "togetherness" of the pair affect the understanding and success. This measure is not constrained only to the moments when there are references (verbal or deictic), but we consider the whole interaction as a ground to measure "togetherness". Furthermore, we will show how can we extend and give feedback based on these findings to another context, from a learning point of view to increase students' engagement.

3 Pair Program Comprehension

3.1 Introduction

In this chapter, we present the analysis of a pair-program comprehension experiment [1] to illustrate the sensitivity of the gaze traces to the different levels of understanding as well as to the different episodes in the interaction. This problem is a two sided coin: it involves the cognitive aspects related to program understanding and the social aspects related to the interaction of two programmers. Through this study, we examine the triumvirate relationship between the gaze, the dialogues and the level of understanding attained by the pair (Figure 3.1).

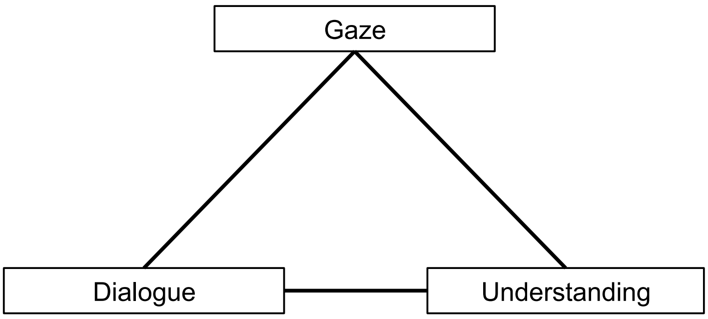


Figure 3.1 – *A typical Diagram to show the relation between the gaze, the dialogues, and the level of understanding of the pair.*

The chapter first describes the context, i.e., pair programming. Then we introduce a few program comprehension strategies that were found in the previous research. Once we have established the context, we provide the details of the experiment and different variables we used to analyse the interaction of the two programmers. Finally, we present the results of the study and the discussion. For this chapter, we conceptualise our domain of investigation as a triumvirate that consists of cognition (program understanding), communication (dialogue), and attention (gaze).

[1] This experiment was conducted by Marc-Antoine Nüssli and Patrick Jermann in June, 2011

3.2 Context

3.2.1 Pair programming

Pair programming [Williams and Kessler, 2000], a method by which the two co-located programmers share a display while performing various programming tasks. The collaborators typically adopt the roles of driver (actual typing) and navigator (focusing on organisational activities and planning) while working. According to the proponents of pair programming, the method leads to higher quality programs in comparison with individual work. More generally, we take pair programming as a special case of collaborative problem solving, a process that involves coordination between participants and the construction of shared understanding.

Pair programming is usually done with co-located programmers. However, spatially distributed pair programming have been studied with satisfactory results showing that the distance factor can be neglected [Baheti et al., 2002]. Pair programming leads to high quality programs [Nüssli, 2011], hence a pair of expert programmers, working in a remote collaborative setting, could obtain a better understanding of a program as well.

3.2.2 Program comprehension as a problem solving task

Program Comprehension is central in many programming tasks, for example during software maintenance or software evolution, where programmers have to read and extend code that they did not necessarily produce themselves. Program comprehension is a special kind of problem solving. Like any problem solving task, program comprehension has a problem statement (to understand the given program) and a solution (the description of functionality of the program) and different approaches to get the solution. The main approaches are top-down and bottom-up. Top-down approach involves decomposition of the problem in sub-problems; and solving the sub-problems, while bottom-up approach involves integration of low-level details to come up with a solution.

Program comprehension is a goal-oriented, problem-solving task that is driven by preexisting notions about the functionality of the given code [Koenemann and Robertson, 1991]. It could be thought of a pattern matching at different levels of abstraction [Tilley et al., 1996]. The different abstraction levels help understanding a program at different levels, for example, at syntactical level programmers could understand the relation between different programming constructs and at semantic level they could relate different programming structures to their real world counterparts. The potential of eye-tracking in diagnosing the quality or the strategies of understanding relies on the assumption that understanding strategies are reflected by different ways to “read” the code.

3.2.3 Program comprehension strategies

There are several strategies to understand a program, a top-down approach [Soloway and Ehrlich, 1984] consists of starting with a hypothesis about the program and then validating or "end marking" the hypothesis with the individual components of the program. A Bottom-up approach [Shneiderman and Mayer, 1979] starts from a series of code fragmentation and then assigns a domain concept to each fragment. An Iterative approach [Brooks, 1983] includes a "while" loop of top-down process, i.e., having a set of preexisting notions or hypothesis, their verification and modification, until everything in the program can be explained within the set of notions with which the iteration started. There are some more strategies that are a hybridisation of top-down and bottom-up [Letovsky, 1987, Von Mayrhauser et al., 1995]. These two strategies are used interchangeably during program comprehension as and when needed [Letovsky, 1987].

Letovsky [1987] proposed a typical set of mental models needed to understand a program which included specific functionality of a program, the way it had beed implemented and relationship among different parts of the program. Letovsky [1987] also emphasised that mental model for implementation consists of actions and data structures of a program. Understanding the entities/data/variables and relationship amongst them inside a program was very important, in order to assign them a concept from the domain knowledge [Biggerstaff et al., 1994]. [Johnson and Soloway, 1985] advocated for having a programming plan to understand the program text (what was written?) and the program intent (why was something written?), and then divided the programming plan into two major parts "Variable plan" (how the data flow of the program worked) and "Control plan" (how were the different conditions related to each other). Johnson and Soloway [1985] then proposed the use of variable plan to understand the relation between program text and program intent.

3.2.4 Elicitations and program understanding

Pennington [1987] gave a special abstract program representation code (control flow, data flow, functional, state charts, condition-action table) to each explanation along with a special knowledge plan code. Each knowledge plan contained a different way to represent the functionality of the given program. For example, the control flow described how the compiler moves between different lines of the program; while the condition-action table listed all the conditions in the program and the how they effected the output of the program. This coding scheme lacked the sense of abstraction hierarchy in the explanation. Having an abstraction hierarchy in the codes is important to know the underlying cognitive (bottom-up or top-down or opportunistic) model of the explanation. Von Mayrhauser et al. [1995] pointed out the need to categorise the dialogues, with each category containing a cognitive significance. The categorisation used by Von Mayrhauser et al. [1995] was too detailed for a program having 100-150 lines of code, as the authors mentioned in that the goal of comprehension, in their case, was software maintenance; for us it provided the basis of understanding through studying the patterns of pairs with good understanding. Good and Brna [Good and Brna, 2004, 2003] gave a coding scheme that is free from program summaries. Their main focus was on finding the information structure produced; and not the underlying cognitive processes in program comprehension.

3.2.5 Expertise and program understanding

A bottom-up approach characterised novice programmers, while experts followed a top-down approach of generating a hypothesis and verifying it in most of the cases. While experts and novices might possess the same semantic knowledge, experts used their experience to make better use of knowledge [Kolodner, 1983].

In two different studies Bonar and Soloway [1983] and Koenemann and Robertson [1991] described the particular strategies for novice and expert programmers respectively. On one hand, Bonar and Soloway [1983] found that for the understanding of novices while loops sometimes become "while demons". Moreover, novices had "conflicts" in the strategies to be applied for giving the "Natural Language Description" of a program. Novices tend to follow the "systematic execution" of the program and increase their chances to get stuck. Line by line understanding is typical in bottom-up integration of program functionality and is characteristic of lack of hypothesis [Bonar and Soloway, 1983].

On the other hand, Koenemann and Robertson [1991] found that experts applied the *as-needed* strategy, where they limited their understanding to only those parts of the program that they find relevant to a given task. Experts did not follow a predefined strategy to understand a program. For example, experts did not decide beforehand to understand a program in "top-down" or "bottom-up" manner. Experts tend to use both of them as and when needed. In another study, Koenemann and Robertson [1991] found that experts used a top-down strategy but, in case of a hypothesis failure a bottom-up strategy was used.

3.3 Problématique

Collaborative interaction consists of a sequence of actions and communicative acts. In order to build models that assess the quality of specific interaction patterns (e.g. is an explanation elaborated or not, was it understood or not), it was necessary to first identify the interaction patterns in the flow of interaction (e.g. when is an explanation given). In order to automatically analyse these interaction episodes we need to find out **how to automatically find interaction episodes** based on raw data streams.

Usually, fixation time is aggregated in predefined areas of interest and researchers report global proportions of attention time dedicated to the different types areas. To measure coupling, cross-recurrence analysis quantifies, as an overall measure, how much the gaze of the collaborators follow each other with a given lag. These fixation based measures aggregate indicators measured in the 100ms range to the whole interaction. The interaction episodes that we proposed to detect on the other hand are situated in between the short time range of a fixation and the long time span of the whole interaction. Figure 3.2 shows the conceptual difference between the fixations and interaction episodes. The main difference is in their respective durations in time and their use to analyse different types of behaviours. This brings us to the main methodological question for the pair program comprehension processes.

Methodological Question What are the different ways to segment the interaction, in a meaningful manner, of a dyad trying to understand a program?

Once we have found the interaction episodes, we addressed the following research question:

Research Question What are the relations among the gaze, the dialogues and the level of understanding of the pair?

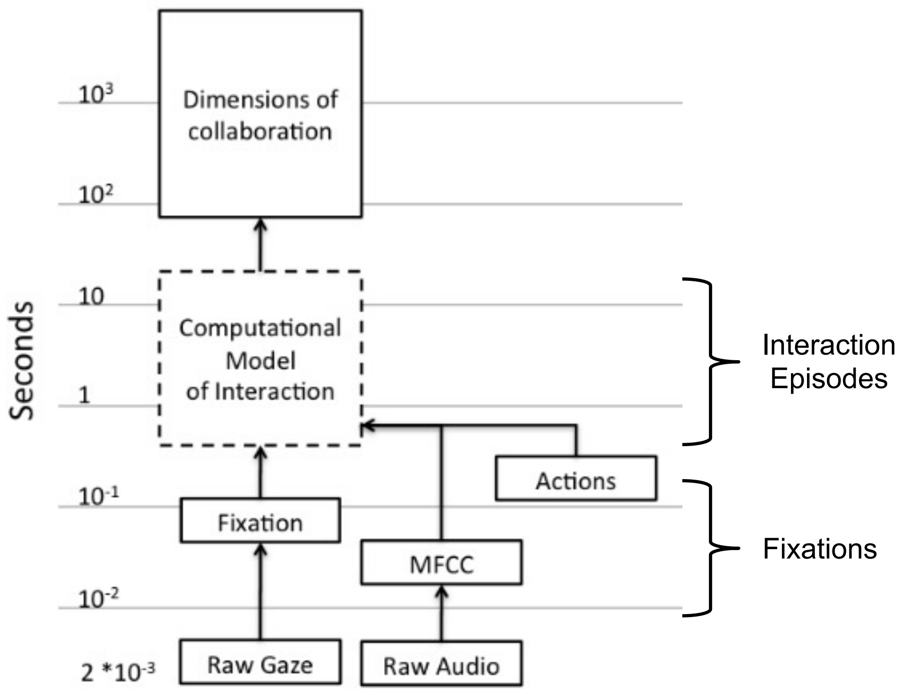


Figure 3.2 – *A typical Diagram to show the conceptual analogy between the fixations and the segments, and to show the analogy between different levels of raw gaze aggregation and the behaviour dimensions*

3.4 Experiment

In the experiment, pairs of subjects had to solve two types of pair programming tasks. The first task was to describe the rules of a game (e.g., initial situation, valid moves, winning conditions, and other rules) implemented as a Java program (Appendix A). The only hint to the pairs was that it is a turn based arithmetic game. The second task was to find errors in the game implementation and to suggest a possible fix using a few lines of output to analyse the error and to find the location of it in the program. For his chapter, we concentrated only on the comprehension task.

3.4.1 Subjects

Eighty-two students from the departments of computer science and communication science from École Polytechnique Fédérale de Lausanne, Switzerland were recruited to participate in the study. They were each paid an equivalent of 20 USD for their participation in the study. The participants were typical bachelor and master students. The participants were paired into forty pairs irrespective of their level of expertise, gender, age or familiarity.

3.4.2 Procedure

Subjects had to read and sign a participation agreement form, when they came to laboratory. Then, for the next 3 minutes, the experimenters calibrated the eye-trackers for each of the subjects. This simple procedure consists of fixating the center of nine circles appearing on the screen. Once both subjects were ready, they individually filled a short electronic questionnaire about their programming skills and previous experience. The pretest which followed, consisted of individually answering thirteen short programming multiple choice questions.

3.4.3 Apparatus and material

Gaze was recorded with two synchronised Tobii 1750 eye-trackers that record the position of gaze at 50Hz in screen coordinates. The eye-trackers were placed back to back and separated from each other by a wooden screen. The synchronisation of the eye-trackers was done by using a dedicated server to log gaze via callback functions from the low-level API of the eye-trackers [Nüssli, 2011]. The subjects heads were held still with an ophthalmologic chin-rest placed at 65 centimetres of the screen. An adaptive algorithm was used to identify fixations and a post-calibration was done to correct for systematic offsets of the fixations with regards to the stimulus [Nüssli, 2011].

The JAVA programs were presented in a custom programming editor based on the Eclipse development environment. Text was slightly larger (18pt) than it is usually on computer screens and was spaced at 1.5 lines to facilitate the fixation hit detection at a word level precision. Scrolling was synchronised between the participants, such that when programmers scrolled, their partners' viewport was also updated at the same time. All other highlighting, search and navigation functionalities were disabled in the editor.

3.5 Variables

3.5.1 Level of Understanding

We distinguished between two levels of understanding based on how well the pair performed the description task. Pairs with high level of understanding were able to describe correctly and completely the rules of the game including initial situation, valid moves and winning conditions. Pairs with low level of understanding could only describe partial aspect of the game structure and tried to guess the detailed rules from the method names; for example, they failed to describe the winning conditions correctly or they explained only some of the initial conditions.

One important point worth mentioning here is that the ratings of levels of understanding are purely based on the correctness of the explanations given by the pairs. For example if a pair gave a description in programming terms (a low level of abstraction) and it was correct, the pair was rated to have a high level of understanding. The reader must not confuse between the program description dialogues (described in section 3.6.3) and the levels of understanding.

3.5.2 Semantic tokens

The program is comprised of tokens. For example, a line of code “location = array [c] ; ” contains 13 tokens (location, c, =, array, ;, 2 brackets and 6 spaces). Fixations on the individual tokens were detected using a probabilistic model (for details see Nüssli [2011]). As the code tokens were small and many in numbers, the probabilities of having a fixation on a token was distributed among several tokens (3 to 10). These probabilities were normalised to make the sum of probabilities for one fixation to be one. We then aggregated the probabilities of all fixations in the defined time window. For each object of interest, the aggregated probabilities were computed as the average of the probabilities of each fixation weighted by the fixation duration. Hence, the resulting aggregate represented a probability distribution over the objects of interest which could be seen as the fixation time ratio based on probabilistic hits values. Finally, we computed the time spent on the various tokens in the program and categorised them into categories named as semantic tokens. For the different analyses, we developed two different versions of this categorisation scheme.

First version with three semantic token categories

Identifier this class included the variable declarations.
Structural this class included the control statements.
Expression this class included the main part of the program, like the assignments, equations, etc.

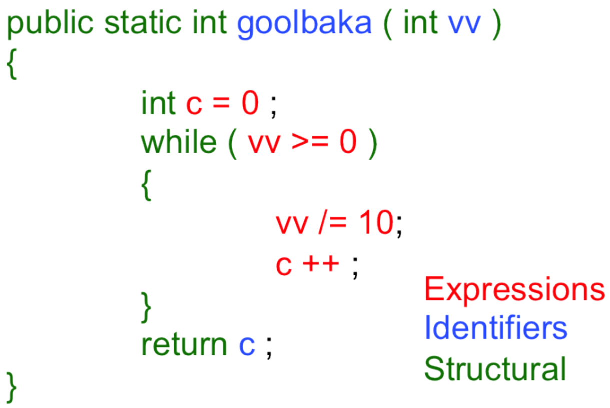


Figure 3.3 – A typical example of semantic elements of a program. The identifiers are the names of the variables and the methods. The structural elements are the punctuation elements and the brackets. The expressions contain the relation among the identifiers.

Second version with six semantic token categories

Structural this class included the control statements.
Type this class included the keywords identifying the data type/structure of a variable or a return data type/structure of a method.
Method this class included names and usage of methods defined by the programmer.
Variable this class included names and usage of variables defined by the programmer.
System method this class included the names and usage of JAVA inbuilt methods.
System variable this class included the names and usage of JAVA inbuilt variables.

3.5.3 Gaze transitions

Is it possible to discriminate the different reading patterns for program understanding between the pairs with high versus low levels of understanding ? Do the pairs with high level of understanding build their understanding based on different semantic elements in the program than the pairs with the low level of understanding?

To measure the reading patterns, one of our approaches was based on gaze transitions between different types of program elements. For defining the gaze transitions we used the semantic tokens the three categories. We proposed that a “back and forth” shift in gaze between identifiers and expressions would depict the attempt to understand the data flow and/or the relation among the variables. Similarly, a gaze shift among all the three semantic classes would translate, in terms of reading patterns, to “Linear reading”.

Our analysis was aimed at finding which type of transitions characterise pairs with different levels of understanding. Table 3.1 shows the categorisation of different transitions among different semantic classes in the program into data flow, control flow and data flow according to control flow. We considered the "3-way" transitions among the semantic classes as one 3-way transition reflected one unit of reading patterns. For example, a 3-way transition "E− >I− >E" reflected the "reference lookup" for a variable in an expression.

Table 3.1 – *Categorisation of different transitions among different semantic classes in the program into different types of flows in the program. (I=Identifier, S=Structural, E=Expression).* $->$ *denotes the transition.*

Type of flow in the program	Types of transitions
Data flow	I− >E− >I E− >I− >E
Control flow	I− >S− >I S− >I− >S
Data flow according to Control flow (Systematic execution of program)	S− >E− >S, E− >S− >E S− >I− >E, E− >I− >S S− >E− >I, I− >S− >E I− >E− >S, E− >S− >I

We followed the following sequence of operations to obtain the transition categories from the raw gaze data:

1) **Raw Gaze and Fixations:** The first step in the analysis of gaze aggregated the gaze points given by the eye tracker into fixations (moments of relatively stable gaze positions).
2) **Determining Areas of Interest or Tokens:** Once we had the fixations from the raw gaze data we define the areas of interest in our stimulus, i.e., in the program.
3) **Episodes of Interaction:** From the fixations we got the interaction episodes using method described in section 3.6.1.
4) **Tokens to Semantic Classes:** After defining the tokens as our areas of interest we used the semantic tokens with three categories (see Section 3.5.2).
5) **Sequence of Semantic Classes looked at:** We took the sequence of the semantic classes fixated during the interaction for our analysis, for example sequence “IIIESSEEESSSIIIE” (I = Identifiers, S = Structural and E = Expressions) tells us that first 3 fixations were on identifiers, 4^{th} fixation was on an expression then next 2 fixations were on the structural elements and so on.
6) **Compressing the Sequence:** As we were interested in the transitions between the semantic classes and not in the duration of time spent on the different semantic classes. We considered the continuous fixations on the same semantic class to be one fixation and thus the sequence “IIIESSEEESSSIIIE” turned into a "compressed" sequence as "IESESIE".
7) **Compressed Sequence to "3 way" Transitions:** Once we had the compressed sequence we simply counted the number of transitions from one semantic class to other and then to another one. For example the compressed sequence "IESESIE" has 5 transitions "IES", "ESE", "SES", "ESI" and "SIE".
8) **Transitions to Control Flow:** Transitions "ISI" and "SIS" depicted the activity of tracing

the control of the program with the different states of the variables.

9) **Transitions to Data Flow:** Transitions "IEI" and "EIE" depicted the activity of tracing the data flow of the program. This reflect the task of looking for different variables and the interdependencies between them.
10) **Transitions to Linear Reading:** All the transitions involving the three semantic classes and the transitions "ESE" and "SES" reflected gaze transition amongst all the semantic elements in a program. This translated to reading the program as if it was an English text.

3.6 Interaction Episodes

In the section 3.3, we highlighted the importance of automatically defining interaction episodes to understand the cognitive mechanisms underlying the pair program comprehension. In this section, we present three methods to define the interaction episodes. The first method used the temporal nature of gaze to define the episodes. The second method used the individual distribution of the gaze over different tokens in the the program and the pair's similarity of this distribution in a given time window. The third method simply used the dialogues to achieve different interaction episodes.

3.6.1 Fixations Episodes

The existence of *fixation episodes* first came to our attention when looking at the evolution in time of the JAVA tokens looked at by the programmers during a program understanding task. The green curve in the figure 3.4 represents the evolution of the average token identifier in time (tokens were numbered in order of appearance in the program) for a particular pair. Stable exploration episodes clearly appear as "plateaux" separated by "valleys" and are reminiscent of the data patterns that characterise the organisation of raw gaze data into fixations and saccades. Deep valleys are due to programmers scrolling through the code while looking for particular methods whereas smaller valleys correspond to focus shifts between areas of program visible on one screen. Computing fixation episodes was a two step process; first we found the individual episodes; and then we aligned them in time to find the interaction episodes for the pair.

Finding segments in the gaze of individual participants

For finding the fixation episodes from individual data, first of all we smooth the fixations using moving averages for each non-overlapping window of 10 seconds.; and then used the following steps to find the segments from the individual fixation data:

1) First, we divided the smoothened fixation data into non-overlapping time windows.
2) For fixations in each window, we found the best fitting line.

3) For each fitted line, we found the angle it made with the time axis; and for each window, we found the range of tokens looked at by the participant.
4) For each window, we found whether the angle between the line and the time axis and the range of tokens looked at were both less than the respective thresholds; if yes, then the window was deemed to be a part of a fixation episode.
5) Once we had the potential portions of a segment; we merged such sequential windows in time, only if they were overlapping in terms of the range of tokens looked at.
6) The output of this step were the fixation episodes for each participant in the pair.

Figure 3.4 shows the episodes computed from the fixation data (sampling rate 50Hz) for two participants in the same pair. The black lines depict the detected episodes. These individual fixation episodes are then aligned in time to find the interaction episodes for the pair. We describe this step in next subsection.

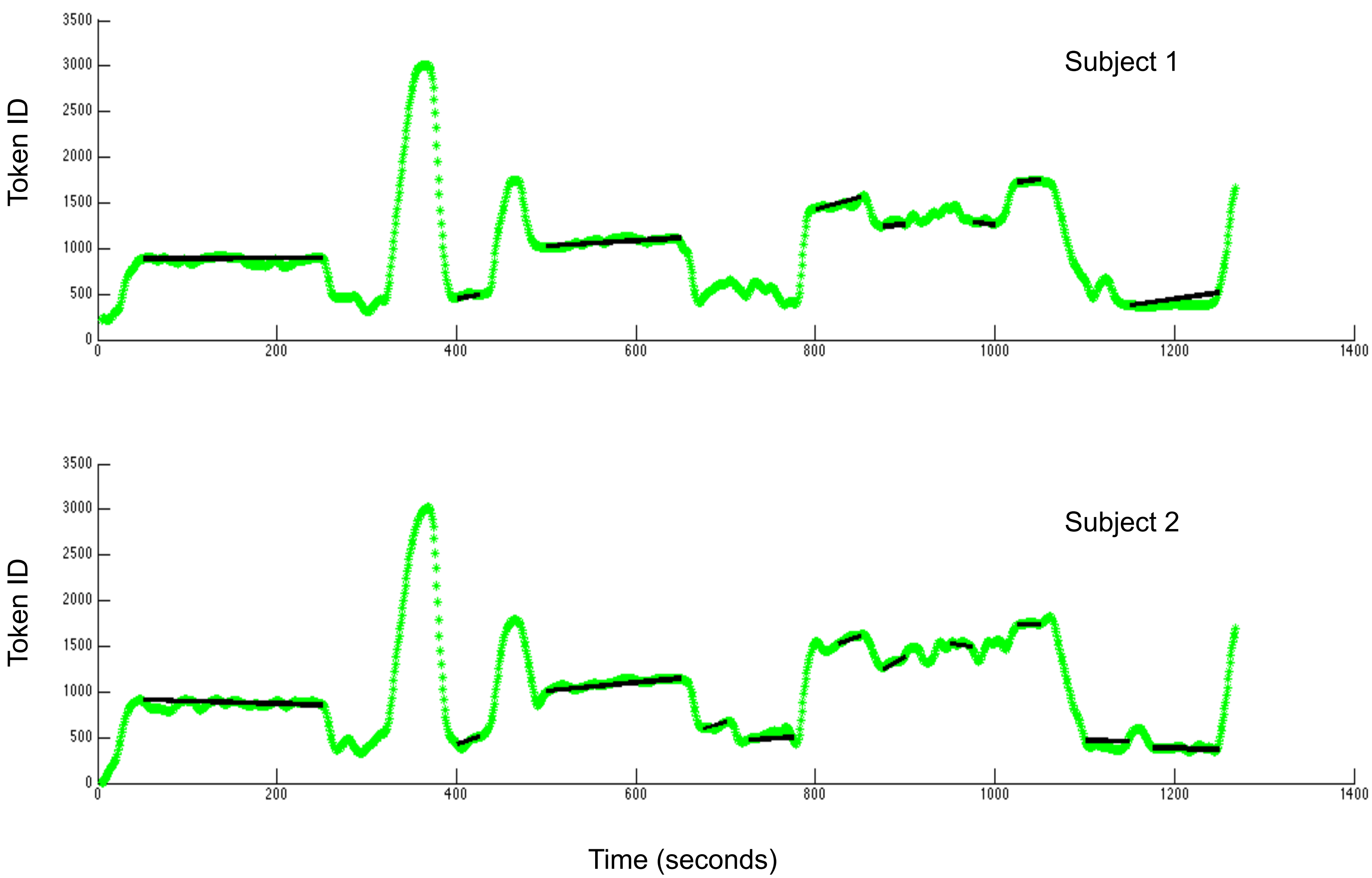


Figure 3.4 – *Fixation episodes computed for individual participants of a pair in the program understanding task. The x axis represents time (sampling rate 50Hz). The y axis represents the average token ID that was gazed at. A horizontal "plateau" (black horizontal lines) means that the subject has been looking at a stable range of tokens over a relatively longer period of time.*

Temporally aligning the episodes for the pair

We aligned individual fixation episodes in time and then again merge them so that we had longer (in terms of time) interaction episodes to analyse.

For finding the interaction episodes, we used the following steps:

1) Input to this step was the two individual episodes that we got as the output of the previous step.
2) We found the temporal overlap between the two individual fixation episodes and created a binary overlap matrix. Each element in this binary matrix indicated whether the i^{th} episode of first participant overlapped (more than a threshold, 60%) with the j^{th} episode of the second participant; both in terms of the time and the range of tokens looked at (intuitively we could say that there is no temporal overlap between the non-consecutive episodes).
3) Once we had the overlap matrix, we considered the intersection of the episodes for the two participants (in terms of their duration) and defined the intersection to be the convergent interaction episodes.
4) The output of this step was the set of convergent interaction episodes for a pair.

Figure 3.5 shows an example of temporal alignment of the individual episodes and the convergent interaction episodes in terms of time.

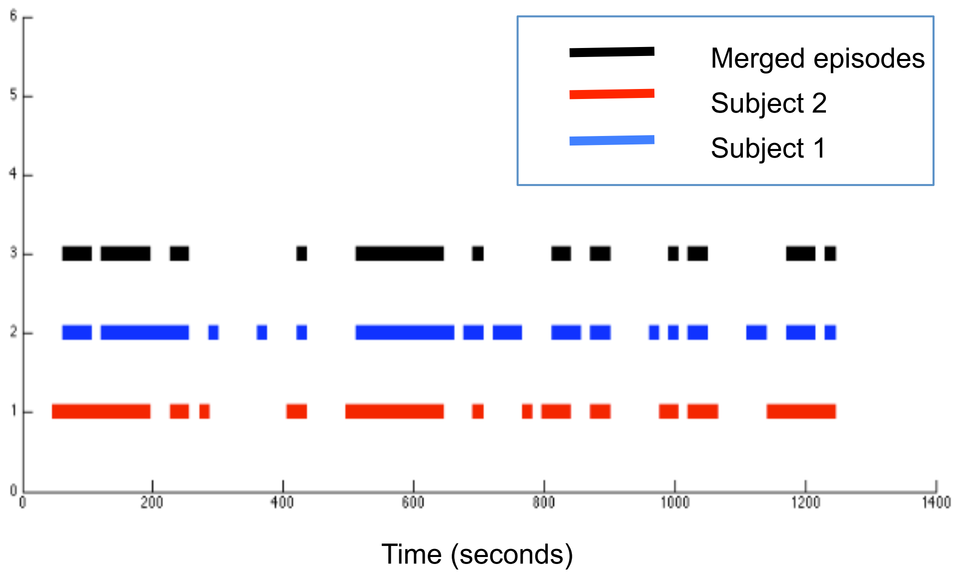


Figure 3.5 – *Fixation episodes of both the participants aligned in time and the episodes of interaction; time on X-axis; Y-axis: 1 for first participant, 2 for the second participant, 3 for the episodes of interaction*

3.6.2 Focus-similarity episodes

The *focus-similarity episodes* were identified based on two parameters: the individual visual focus of gaze and the pair's gaze similarity. In order to characterise the individual visual focus of each subject, we computed the object density vector over a given time window. This density vector contained the probability of looking at the different objects of the stimulus. In order to compute this vector, we aggregated gaze data over a 1-second time window and we compute for each object the amount of gaze time that was accumulated inside the object.

We then defined the *individual visual focus size* (Figure 3.6) as the numbers of objects that are looked at during a 1-second time frame. The rationale was to distinguish between moments where the subjects looked essentially at few objects versus moments where they looked almost uniformly at several objects. In order to get a quantitative indicator of this focus size, we computed the entropy of the density vector. Entropy measures the level of uncertainty of a random variable, which, in our case, was the number of objects looked at by the subjects. Hence, high entropy indicated that the subjects looked at many objects (not focused gaze), while low entropy indicated that they mostly looked at few objects (focused gaze).

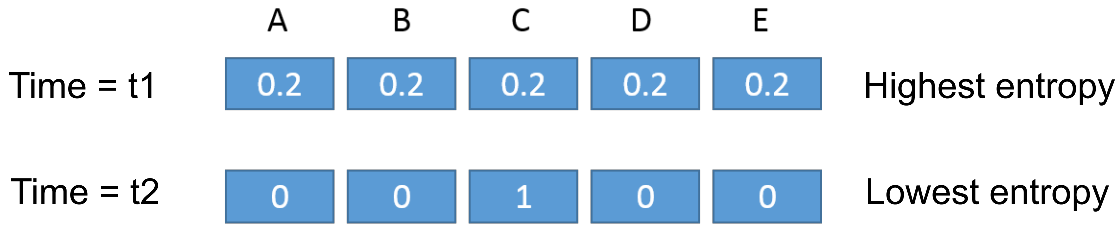


Figure 3.6 – *A typical example of computing gaze entropy for an individual. The letters are symbolic semantic tokens. The numbers inside the boxes represent the proportion of the time window spent on the respective semantic tokens. We show the two extreme cases with highest an lowest possible values of entropy.*

Next, for each 1-second timeframe, we defined the *pair's visual focus coupling* (Figure 3.7) as the *similarity* between the objects looked by one subject and the objects looked by the second subject. We quantified this coupling by computing the cosine between the gaze density vector of one subject and the gaze density vector of the other subject.

The *focus-similarity episodes* were obtained by combining focus size and similarity. An episode lasted as long as the *individual focus size* and the *pair's similarity* stayed constant. Technically, a run length encoding procedure applied on the 1-second indicators for the visual focus and the similarity obtained this. When both subjects were focused and similar we defined **"focused together"** gaze episodes. Similarly, we defined the other three types of gaze episodes that were: *1)* **"not focused together"**, *2)* **"focused not together"**, and *3)* **"not focused not together"**. Since we were mostly interested in *"what happens during moments of high togetherness?"* we report only what happened in "together" episodes (i.e., "focused together" and "not focused together"). Typically, a "focused together" episode translated in terms of behaviour as putting

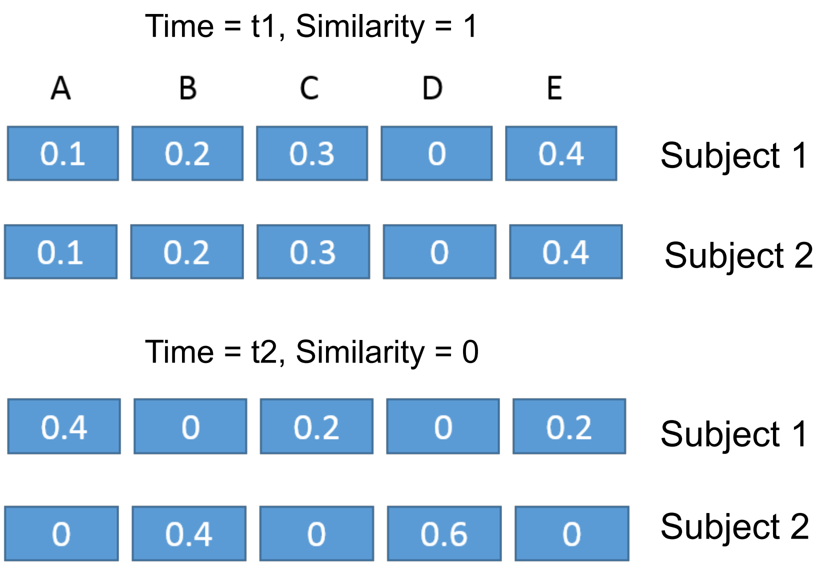


Figure 3.7 – *A typical example of computing gaze similarity for a pair. The letters are symbolic semantic tokens. The numbers inside the boxes represent the proportion of the time window spent on the respective semantic tokens. We show the two extreme cases with highest an lowest possible values of gaze similarity.*

joint efforts to understand code; while a “not focused together” episode translated as an effort to search some piece of code.

3.6.3 Dialogues

Dialogues in a collaborative program understanding task help us to identify various collaborative activities (controlling the scroll, managing time and task) and descriptions which could be used to find the interaction episodes for further gaze analysis.

The categorisation schemes, described here, was developed to account for the program descriptions done by individual programmers. In a pair programming setup, collaboration also plays an important role apart from individual efforts to understand the program. None of the three coding schemes ([Pennington, 1987, Good and Brna, 2004, 2003], presented in section 3.2.4) had categories that could address collaboration in a pair program comprehension task.

We developed a new categorisation scheme, that not only considered the description dialogues, but also the collaborative activities involved. This scheme characterised code descriptions in terms of both the scope and the abstraction of the program description. The categories were well suited for programs with 100-150 Lines of Code(LOC) and they could be used to reflect the mental processes (top-down or bottom-up) underlying the program comprehension activities. For categorising the dialogues, we transcribed the audio recordings from the pairs. There were only 16 pairs who talked in English, hence, we will show the results for only those pairs.

We divided the dialogues into 2 main categories: *program description* and *collaboration management*. First 4 categories contained the dialogues to identify the different descriptions of the program and later 4 categories contained the dialogues for collaboration management activities. The program description dialogues could further be categorised as a two-dimensional scheme as shown in table 3.2. On one dimension there is the level of abstraction in the explanation of the program. On the second dimension there is the length of the program that was explained in terms of Lines of code (LOC). Such representation also helped interpreting in terms of different program understanding strategies according to description dialogues within the table. For example, given a series of dialogues, moving right or moving up in the grid would be interpreted as Bottom-Up and moving left or moving down would be interpreted as Top-Down. Readers might think that the program description dialogues simply reflected the process of assigning each pair a level of understanding; this is not the case. The level of understanding was assigned based on the correctness of the description and not on based on how abstract the description was or how big the part of program was covered. The dialogue categories are explained as following.

1) **Program Description Dialogues (DESC)**
METH_OPR Description in programming terms for a scope of one line of code. Example, "while $winner = 0$ and not gameFinished $currentPlayer = 2 - currentPlayer + 1$".
METH_ACT Description in programming terms and in english for a scope of 2 to 10 lines of code. Example, "when the game is not finished and there is no winner it continues you go to the next player."
LINE_OPR Description in programming terms for one line of code. Example," choice is getPlayerMove currentPlayer".
LINE_ACT Description in programming terms and in english for a scope of 2 to 10 lines of code. Example, "player makes his choice with getPlayerMove".

2) **Collaboration Management Dialogues (MGMT)**
TM Overall Management of the task the participants did inside the phases and questions. Reading instruction, reading questions, talking about remaining time, deciding to answer. Example, "Let's start recording answer."
TMT Group of Task Management statements that depicted the order of the tasks that were to be done during the experiment. In other words, this category captures the meta level dialogues about the procedure. Example, "Lets starts the phase, I'll read the questions."
FM Managing the Focus of the gaze during the task. Talking about navigation. Telling where to look at. Asking where something is. Example, "Where is the function checkAndSet?"
TECH Any dialogues related to the controls of interface, scrolling, view-port, discussions about how selection sharing works. Example, "When you scroll it moves for me too".

Other description measures derived from table 3.2 were the "scope" and the "abstraction" of

Table 3.2 – *Examples of program description dialogues (Excerpts from the audio transcriptions).*

		Scope of the program described	
		One line of the program	2-10 lines of the program
Abstraction in the description	Low	player makes his choice with getPlayerMove	while winner = 0 and not gameFinished currentPlayer = 2 − currentPlayer + 1
	High	choice is getPlayerMove currentPlayer	when the game is not finished and there is no winner it continues you go to the next player.

the description. Scope and abstraction were calculated with adding the rows and columns of the table 3.2 respectively.

3.7 Results

Once we solved our methodological question, we moved ahead and tried to find the answers to our research questions. We analysed the whole interaction from three different perspectives[2]: *1)* the gaze transitions for different fixation episodes; *2)* the distribution of gaze over different semantic tokens during different dialogue episodes; *3)* The interlacing of gaze and dialogue episodes to analyse the interaction over different time granularities.

3.7.1 Temporal interaction

We found a relation between the *level of understanding* of the pair (U), the *pair composition* (P) and the *gaze transitions* (T) using log linear models [Gottman and Roy, 1990]. Log linear models use contingency tables to find the relation between different variables and for comparing the two models for same contingency table. Gottman and Roy [1990] used a new statistics, called G^2 the "likelihood statistics" (or LRX^2), which is asymptotic to "chi square". G^2 can be calculated as following:
$G^2 = 2\sum_i (observed)_i log \frac{(observed)_i}{(expected)_i}$

There are two main methods for fitting the log linear model to a given contingency table. *Forward Selection*, where we fit all hierarchical models that include the current model and differ it by one effect (single or interaction effect); and *Backward Elimination*, which leaves the term that incurs the least change in the LRX^2 value (for details see Gottman and Roy [1990]). We combined both of the methods to achieve a fast consensus. According to the forward selection, we fitted all the hierarchical models that differ the current model by one term. For the next iteration, we kept the model with the least change in the LRX^2 value (opposite to the

[2]For the first two perspectives, we only compared the very distinct pairs, i.e., 16 and 12 pairs for high and low levels of understanding respectively. For the relation between the gaze and dialogues, we had to transcribe the audio from the pairs, we only transcribed those who spoke English, hence we had 8 pairs in both high and low levels of understanding.

backward elimination, but the idea was to delete the least change incurring term). The finally selected model should have the maximum degrees of freedom with the least change in the "likelihood statistics" (or LRX^2).

Table 3.3 – *Hierarchical linear model fitting for Contingency Table with dimensions Transition (T), Pair Type (P) and Level of Understanding (UND), for the combined gaze of all the pairs*

Model	G^2	**DoF**	**Terms Deleted**	$\triangle G^2$	$\triangle DoF$
$[T][P][U]$	7503	57			
$[TPU]$	0	0			
$[TP][TU][PU]$	32	22	$[TPU]$	32	22
$[TP][TU]$	7267	24	$[PU]$	7235	2
$[TP][PU]$	103	33	$[TU]$	71	11
$[TU][PU]$	54	44	$[TP]$	22	22
$[TU]$	8026	48	$[PU]$	7972	4
$[PU]$	8487	66	$[TP]$	8433	22

Table 3.3 shows the log linear model fitting using the method proposed above. The first 2 models $[T][P][U]$ and $[TPU]$ are the *independence model* and the *saturated model* respectively. We can see that the saturated model fitted the data perfectly ($DoF = 0$, $G^2 = 0$). On the other hand, independence model showed a big variation ($DoF = 7503$, $G^2 = 57$) from saturated model. Removing the 3-way interaction term resulted in the model $[TP][TU][PU]$ ($DoF = 32$, $G^2 = 22$). Now, we considered the effect of removing one 2-way interaction term at a time. Removing term $[PU]$ caused a big deflection from the all 2-way terms model with a small increase in the degrees of freedom ($\triangle G^2 = 7235$, $\triangle DoF = 2$). Removing $[TU]$ also caused some deflection from the all 2-way terms model ($\triangle G^2 = 71$, $\triangle DoF = 11$); but removing $[TP]$ term caused the smallest deflection and increases the degrees of freedom as well ($\triangle G^2 = 22$, $\triangle DoF = 22$). Further removing terms from $[TU][PU]$ caused greater deflections. The best fit model for a given contingency table is the one with the least $\triangle G^2$ and largest $\triangle DoF$ with respect to the saturated model ($[TU][PU]$ in this case with $\triangle G^2 = 22$ and $\triangle DoF = 22$).

It was clear from finally selected log linear model that there was a dependence between the *transitions* and the *level of understanding*; as well as between the *pair type* (the pair composition in terms of experts and novices) and *level of understanding*.The first dependency is reflected by the term $[TU]$ and the later one was depicted by the term $[PU]$, where $[PU]$ reflected the fact that a pair of two experts could understand a program better than a pair of two novices. To better understand the dependency between transitions and levels of understanding we used ANOVA. Here, instead of using the transitions, we grouped them in categories as depicted in table 3.1. Figure 3.8 shows the differences between the two levels of understanding (medium and high) for the different types of flows in a program.

Pairs with low level of understanding had significantly more transitions amongst all three semantic classes than the pairs with high level of understanding ($F[1,27] = 32.1, p < .01$). In terms of gaze transitions; this behaviour translated into reading each line of the program and trying to understand it. This showed that these pairs looked simultaneously at the conditions

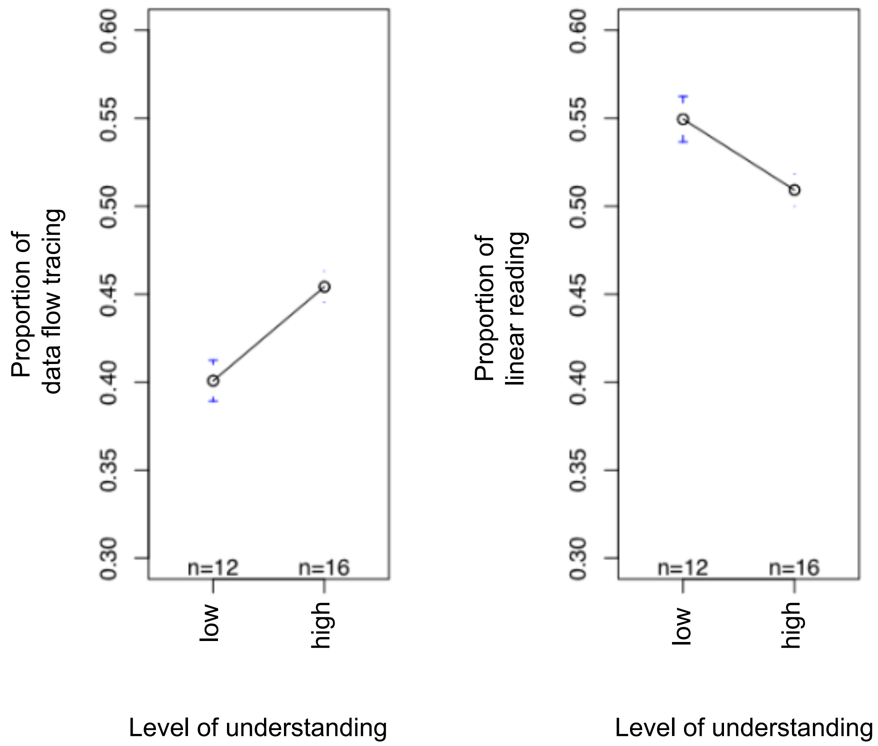


Figure 3.8 – *Mean plots and confidence intervals for different transitions for the whole interaction.*

in the program as well as the modification of the data elements according to the conditions. This strategy of program understanding was similar to reading the program line by line; in other words, reading the program as if it was an English text. This method was not characteristic of the pairs with high level of understanding, as shown in an experiment by Koenemann and Robertson [1991]. **The pairs with high level of understanding had significantly more transitions among the identifiers and the expressions** ($F[1,27] = 65.5, p < .01$). They concentrated more on the variable/entities and the relationship among them. Building up their understanding in this manner the pairs with the higher level of understanding were able to do a proper concept assignment from the program domain to the world domain as shown by Biggerstaff et al. [1994].

Taking our analysis one step ahead, to find the effect of the convergent and divergent episodes of interaction, we carried out 2 × 2 ANOVA for data flow and systematic execution with two factors level of understanding and convergent/divergent interaction episodes.

Table 3.5 shows the descriptive statistics for the proportion of data flow transitions and the linear reading transitions in the different types of interaction episodes and for the different levels of understanding. There were two single effects for the type of interaction episode ($F[1,27] = 121, p < 0.01$) and for the levels of understanding ($F[1,27] = 10.86, p < 0.01$), and there was no interaction effect. From figure 3.9 we observed that **all the pairs in divergent**

Table 3.4 – *Mean and standard deviations for the data flow and the linear reading across the two different levels of understanding.*

Transition type	Low level of understanding ($n = 12$)		High level of understanding ($n = 16$)	
	Mean	Std. dev.	Mean	Std. dev.
Data flow	0.4	0.018	0.45	0.016
Linear reading	0.55	0.020	0.51	0.017

phases of interaction spent more time on understanding the data flow than that in convergent phases. Moreover, from figure 3.10 we observed that **the pairs in convergent phases have a had higher ratio of transitions that correspond to linear reading**. There is also an effect of levels of understanding on systematic program execution depicting more effort put by the pairs with medium level of understanding on systematic program execution.

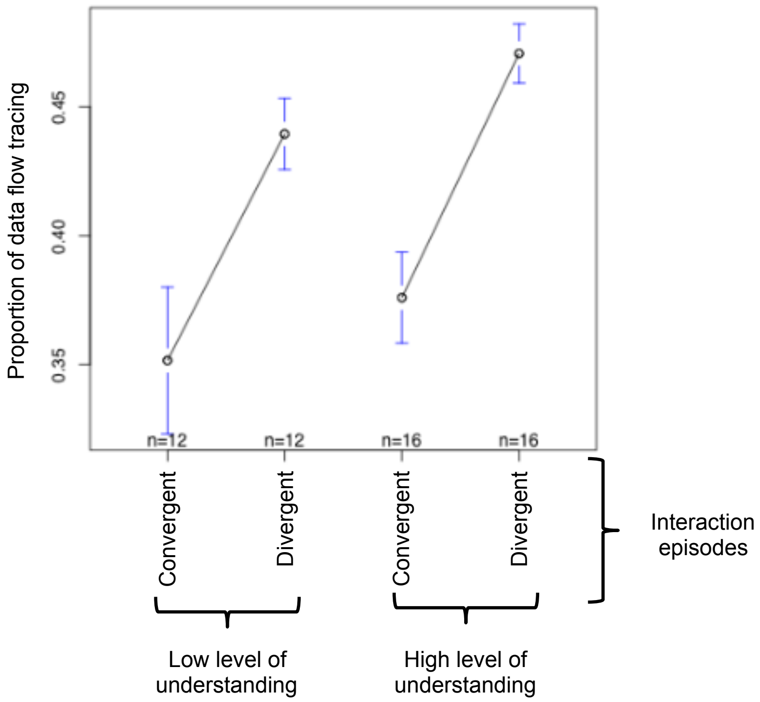


Figure 3.9 – *Mean plots and confidence intervals for data flow for the interaction episodes and levels of the understanding.*

Using the results from the fixation episodes and the gaze transitions, we could say that there was a significant relation between the levels of understanding attained by the pair and their gaze patterns (figure 3.11). The pairs with high level of understanding followed the data flow of the program and the pairs with low level of understanding read the program as if it was an English text.

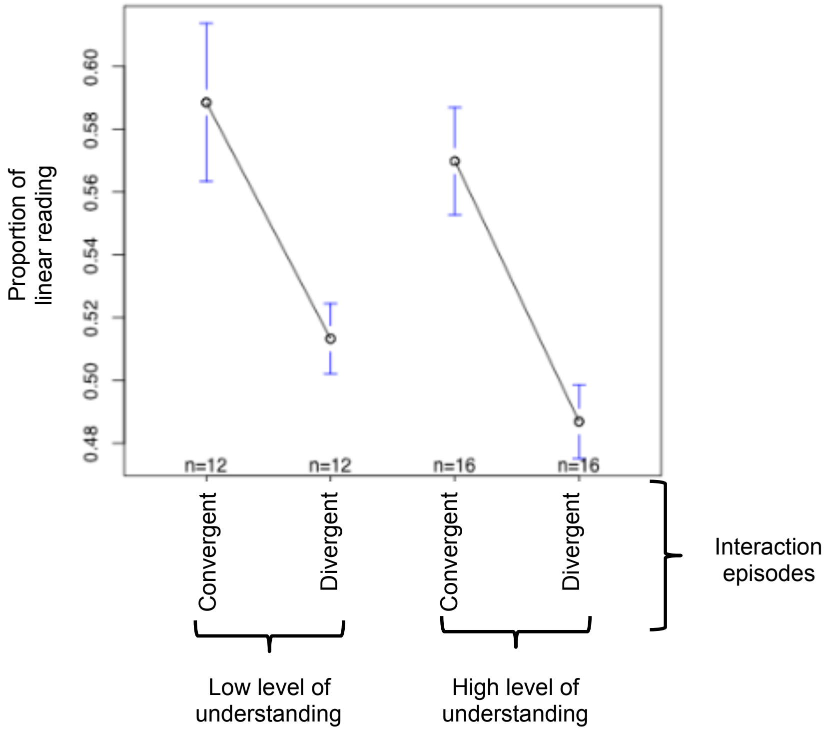


Figure 3.10 – *Mean plots and confidence intervals for Systematic reading of program for the interaction episodes and the levels of understanding.*

Table 3.5 – *Proportions of data flow and linear reading transitions (mean and standard deviation) by type of episode and level of understanding.*

		Level of understanding	
Transition Type	**Episode Type**	**Low** (n=12)	**High** (n=16)
Data Flow	Convergent	0.59 (0.04)	0.56 (0.03)
	Divergent	0.51 (0.02)	0.49 (0.02)
Systematic Execution	Convergent	0.35 (0.04)	0.38 (0.03)
	Divergent	0.44 (0.02)	0.47 (0.02)

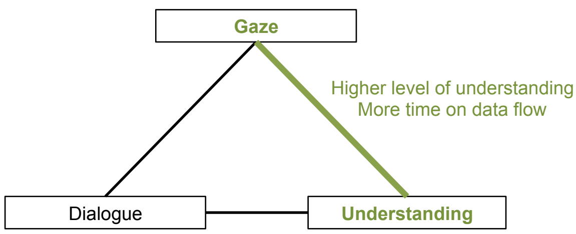


Figure 3.11 – *Contribution to the triumvirate relationship between the gaze, the dialogues and the level of understanding (figure 3.1) from analysing the temporal interaction.*

3.7.2 Gaze-dialogue coupling

Next, we looked at the relationship between the abstraction in program descriptions given by participants and gaze-base descriptors. We first present the log linear analysis for three variables: *semantic token* (C), *scope of description* (S) and *abstraction in description* (A). Once we had dependencies then we present the descriptive statistics to explain the dependencies.

Table 3.6 – *Hierarchical linear model fitting for Contingency Table with factors semantic token (C), abstraction in description (A) and scope of description (S), for the combined dialogues of all the pairs.*

Model	DoF	G^2	Terms Deleted	$\triangle DoF$	$\triangle G^2$
$[C][A][S]$	16	142			
$[CAS]$	0	0			
[CA][CS][AS]	**5**	**6**	**[CAS]**	**5**	**6**
$[CA][CS]$	6	127	$[AS]$	1	121
[CA][AS]	**10**	**12**	**[CS]**	**5**	**6**
$[CS][AS]$	10	20	$[CA]$	5	14

Table 3.6 shows the log linear model fitting using the method proposed in the previous subsection. The first 2 models $[C][A][S]$ and $[CAS]$ are the *independence model* and the *saturated model* respectively. We can see that the saturated model fitted the data perfectly ($DoF = 0$, $G^2 = 0$). On the other hand, independence model showed a big variation ($DoF = 16$, $G^2 = 142$). Removing the 3-way interaction term resulted in the model $[CA][CS][AS]$ ($DoF = 5$, $G^2 = 6$). Now, we considered the effect of removing one 2-way interaction term at a time. Removing term $[AS]$ caused a big deflection from the all 2-way terms model with a small increase in the degrees of freedom ($\triangle G^2 = 121$, $\triangle DoF = 1$). Removing $[CA]$ also caused some deflection from the all 2-way terms model ($\triangle G^2 = 14$, $\triangle DoF = 5$); but removing $[CS]$ term caused the smallest deflection and increases the degrees of freedom as well ($\triangle G^2 = 6$, $\triangle DoF = 5$). Further removing terms from $[CA][AS]$ caused greater deflections.

We can see in Table 3.6 that $[CA][AS]$ was the closest to the model having all the two way interaction terms and thus closest to the saturated model. Hence, we could take this model as our fit. According to this model we could say that there was a dependence between *semantic tokens* and the *abstraction in description* as well as between the *scope of description* and *abstraction in description.* To better understand these dependencies, we used the chi square test.

Code level abstraction was accompanied with the gaze on semantic token "system_method" and high level abstraction was characterised by "method" ($\chi^2(N = 953) = 20, p = 0.001$). Table 3.7 shows the semantic tokens looked at for the different levels of abstraction. We observed that the semantic tokens were related to abstraction in a similar way as with the level of understanding. The reason for a similar relation could be explained by the fact that abstraction in description was closely related to the level of understanding. **Pairs with low level**

of understanding had code level abstraction in description while pairs with high level of understanding had high level abstraction in description ($F[1,953] = 30, p < .001$).

Table 3.7 – *Semantic tokens looked at for different levels of abstraction. Numbers in parentheses are standardised chi square residuals. Residuals (absolute values) bigger than 1.96 are considered statistically significant.*

	Abstraction in Description	
Semantic tokens	Code Level	High Level
Structural	16 (0.70)	25 (0.49)
Method	75 (-2.48)	230 (1.73)
System_method	59 (2.58)	70 (-1.80)
Variable	144 (0.44)	280 (-0.30)
System_variable	6 (-0.33)	15 (0.23)
Type	12 (0.36)	21 (-0.25)

Table 3.8 shows the relation between the scope of program description and the abstraction in description ($\chi^2(N = 953) = 112, p < 0.001$). We observed that often the description for one line of program had the code level abstraction and the description for a bigger scope had the high level of abstraction. One reason for this fact could be that, to have a high level of abstraction one needs to attain a certain level of understanding that is very difficult to get from one line of code.

Table 3.8 – *Scope of description vs. Abstraction in description. Numbers in parentheses are standardised chi square residuals. Residuals (absolute values) bigger than 1.96 are considered statistically significant.*

	Abstraction in Description	
Scope of Description	Code Level	High Level
LINE	192 (7.2)	157 (-5.52)
METH	120 (-5.07)	484 (3.85)

The results from analysing the gaze-dialogue coupling, during pair program comprehension task, suggested that there was a strong relation between the gaze patterns and the dialogues (figure 3.12). The high level of abstraction in the dialogues was accompanied by participants looking at the different parts of program than in the case of low level abstraction in the dialogues. Moreover, the level of understanding was also observed to be significantly related to the level of abstraction in the dialogues (figure 3.12). The pairs with high level of understanding had more abstraction in their dialogues than the pairs with low level of understanding.

3.7.3 Combining gaze, dialogues and understanding

Once we established the gaze-dialogue and gaze-understanding relations, next step was to combine the three variables. For this purpose we divided the whole interaction in task, unit task and operation levels (Figure 3.13).

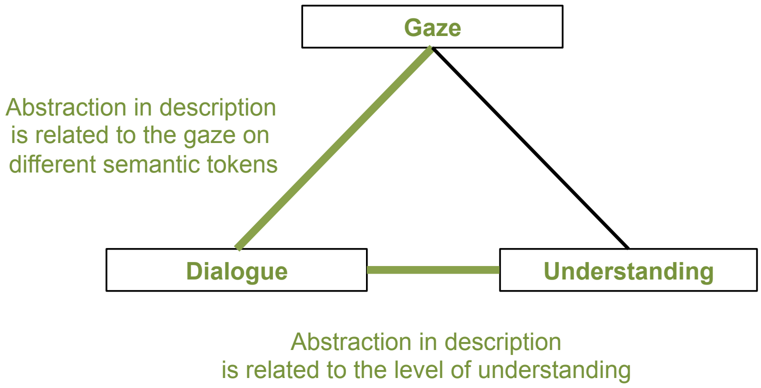


Figure 3.12 – *Contribution to the triumvirate relationship between the gaze, the dialogues and the level of understanding (figure 3.1) from the gaze-dialogue coupling.*

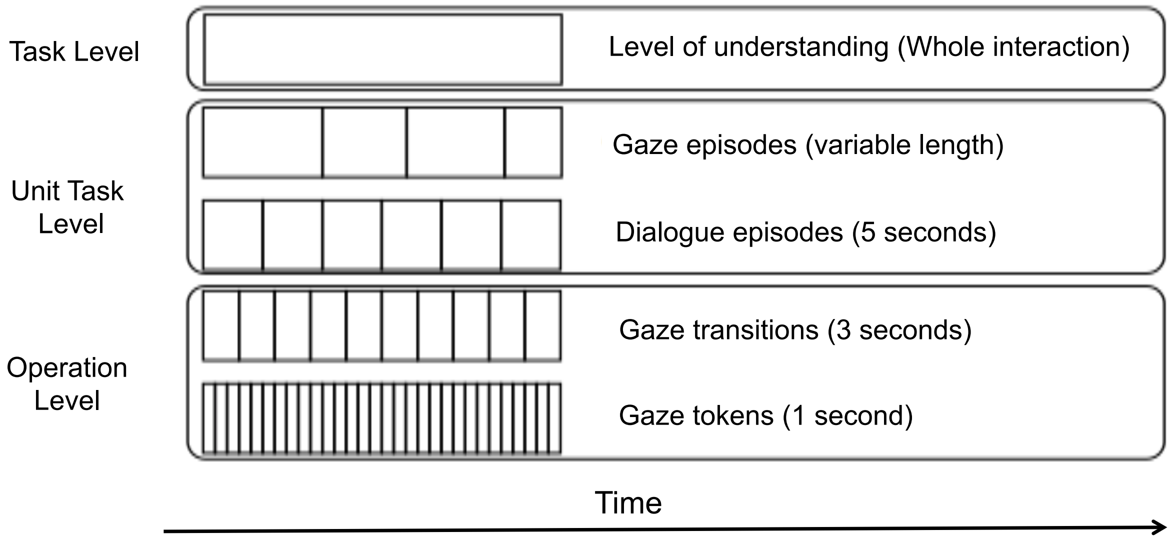


Figure 3.13 – *Interaction of the pair divided into different levels of time granularities.*

- On the **task level**, we rated the level of understanding based on the explanations that were provided by the participants.
- On the **task unit level**, focus-similarity episodes corresponded to moments characterised by a focus-similarity episodes. For example, in a focused-together episode, programmers looked together at a limited set of objects. These episodes typically last from 5 seconds up to 100 seconds.
- On the **task unit level**, we categorised the dialogues of participants depending on whether they were describing the program, or whether they were about managing the task.
- On the **operations level**, we used gaze transitions among different set of objects. The

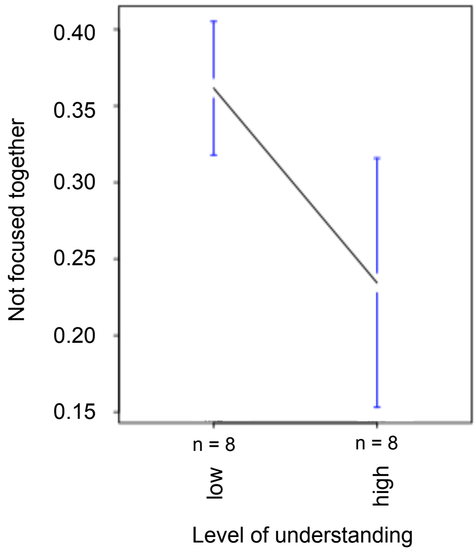


Figure 3.14 – *Mean plots and confidence intervals for not focused together episodes for different levels of understanding.*

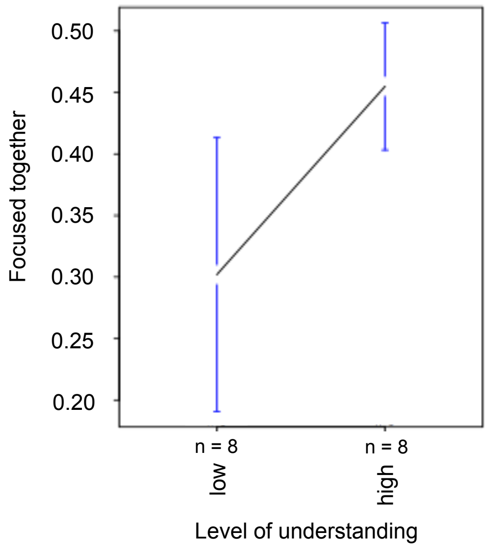


Figure 3.15 – *Mean plots and confidence intervals for focused together episodes for different levels of understanding.*

transitions were based on a segmentation of gaze into 1-second slots and last for 3 seconds.

The first relation was between the *level of understanding* attained by the pair and proportion of time spent by the pair in the different *focus-similarity episodes*. Table 3.9 shows the ANOVA results for gaze episodes "focused together" and "not focused together" across the two levels of understanding. **Pairs with high level of understanding spent more time in gaze episode "focused together" than the pairs with low level of understanding** (F [1,15]=7.580,p=0.01). Figures 3.14 and 3.15 show the mean plots for the two types of gaze episodes across the levels of understanding.

Table 3.9 – *Means and standard deviations for different gaze episodes across two levels of understanding.*

	Low level of understanding (n=8)		**High level of understanding** (n=8)	
Episode type	Mean	Std. dev.	Mean	Std. dev.
Focused together	0.29	0.16	0.46	0.07
Not focused together	0.36	0.07	0.23	0.09

Next, we addressed the relationship between the *focus-similarity episodes* and the *dialogue episodes*. Table 3.10 shows the mixed effect model for the two types of dialogue episodes with the factors level of understanding (UND) and focus-similarity episodes (EPGAZE). There was no significant difference between the proportion of total time spent in dialogue episodes and the gaze episodes, but, there was a significant interaction effect of level of understanding and gaze episodes on the proportion of total time spent on the different dialogue episodes (F [1,61]=7.60, p=0.01, Figures 3.16 and 3.17).

Table 3.10 – *Mixed effect model for dialogue episodes with factors level of understanding (UND) and focus-similarity episodes (EPGAZE) (NS= Not Significant).*

	Dialogue Episodes							
	Description Episodes				Management Episodes			
Model	Df	Sum Sq.	F-value	p-value	Df	Sum Sq.	F-value	p-value
UND	1	0.05	2.46	NS	1	0.01	1.56	NS
EPGAZE	1	0.04	1.71	NS	1	0.01	0.52	NS
UND*EPGAZE	1	0.17	7.80	0.009	1	0.07	7.60	0.01

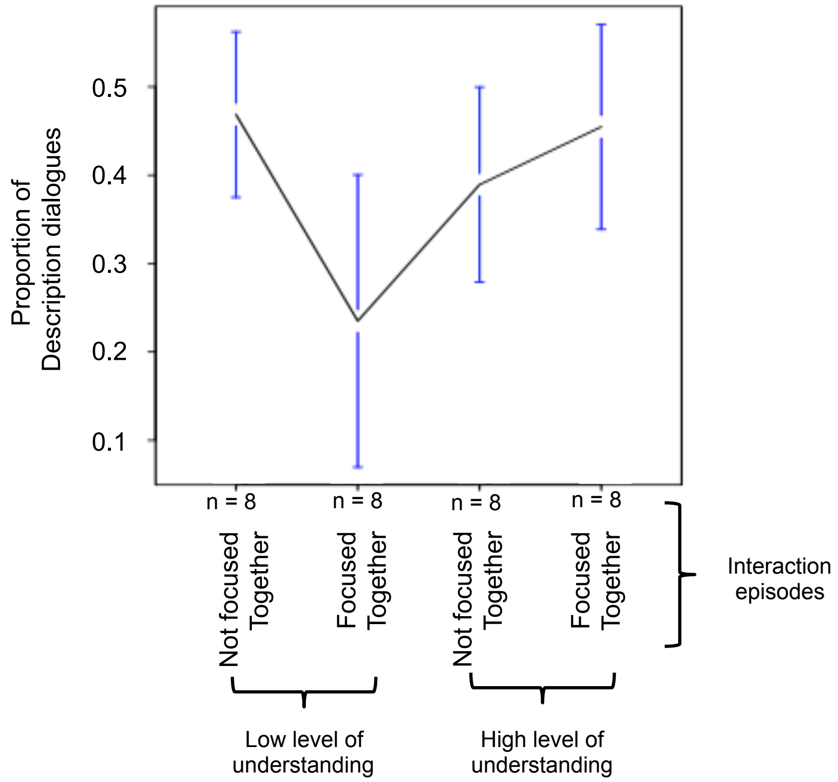


Figure 3.16 – *Interaction effect on DESC (description) dialogues in focused together and not focused together episodes for different levels of understanding.*

The pairs with high level of understanding spent more time in "description" dialogue episodes when they are in a "focused together" gaze episode. On the other hand, **pairs with low level of understanding spent more time on "management" dialogue episodes when they are in a "focused together" gaze episode**. Table 3.11 shows the dialogue snippets for pairs with different levels of understanding during different gaze episodes.

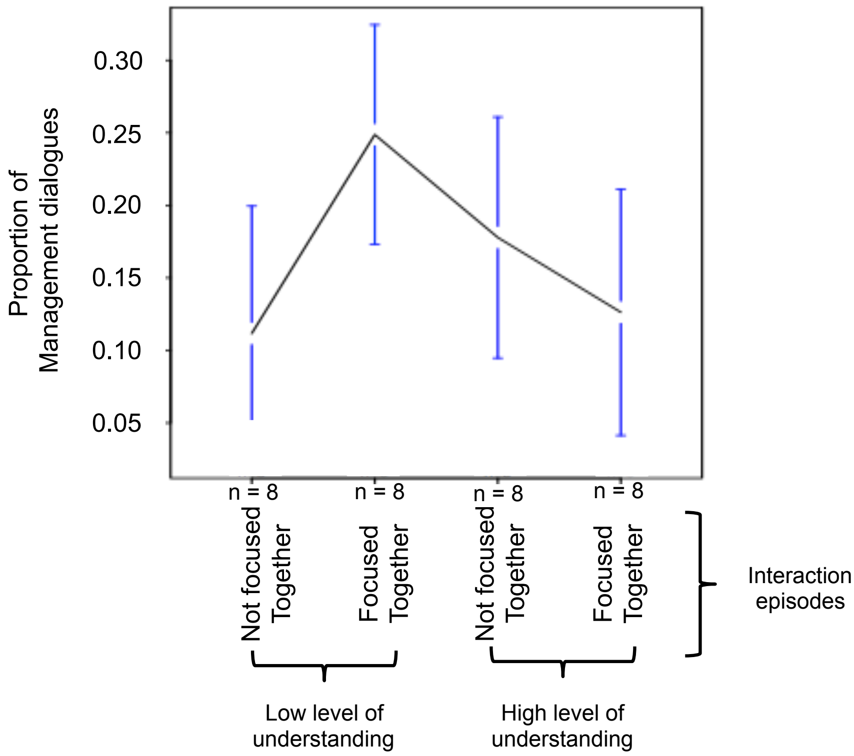


Figure 3.17 – *Interaction effect on MGMT (management) dialogues in focused together and not focused together episodes for different levels of understanding.*

Table 3.11 – *Dialogue snippets for pairs having different levels of understanding during different gaze episodes to show the differences between verbal communications.*

	Focused together	**Not focused together**
Low level of understanding	s2: I'm looking for checkForWinner... the checkForWinner calls the checkForSum function for all i1, i2, i3	s1: look here at the choice... s2: but we don't know where getPlayerMove is... s1: where is getPlayerMove! s2: look here choice is getPlayerMove
High level of understanding	s1: we said before, to be a valid action the player should choose a number which is valid, so from 1 to 9... if initial state or he should choose from the number from the available list	s1: we should look at the current situation s2: currentGameState... s1: no no no... let's check the checkForWinner function

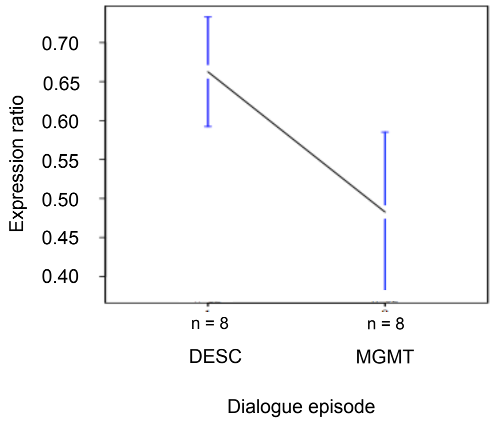


Figure 3.18 – *Mean plots and confidence intervals for "expression" gaze transitions for different dialogue episodes.*

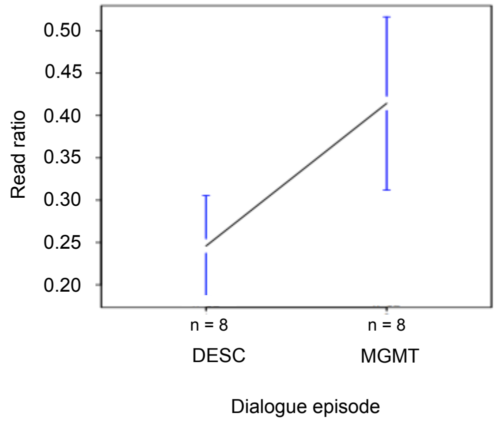


Figure 3.19 – *Mean plots and confidence intervals for "read" gaze transitions for different dialogue episodes.*

Finally, we considered the relation between the *dialogue episodes* and the *gaze transitions* (figure 3.20). Table 3.12 shows the mean and standard deviation values for the different gaze transitions across different dialogue episodes. **"Description" dialogue episodes had more gaze transitions as "expressions" than the "management" dialogue episodes** ($F[1,15] = 8.79, p < .01$). Moreover, **"management" dialogue episodes had more gaze transitions as "read" than the "description" dialogue episodes** ($F[1,15] = 8.31, p < .01$). The differences were irrespective of the level of understanding or the type of gaze episodes. Figures 3.18 and 3.19 show the mean plots for the two gaze transition categories across the different dialogue episodes.

Table 3.12 – *Mean and standard deviations for the different gaze transitions across the different dialogue episodes.*

	Description dialogues		**Management dialogues**	
Gaze transition	Mean	Std. dev.	Mean	Std. dev.
Expression	0.67	0.10	0.48	0.15
Read	0.25	0.14	0.43	0.09

3.8 Discussion

In the previous sections, we presented the methods for and results from analysing the pair program comprehension from three different perspective. In this section, we present the plausible explanations for the results we found.

The first perspective was concerned about the relation between understanding, gaze transitions and the convergence (fixation-episodes) in the interaction. It appeared that the gaze of pairs who understood the program better transition more frequently between identifiers

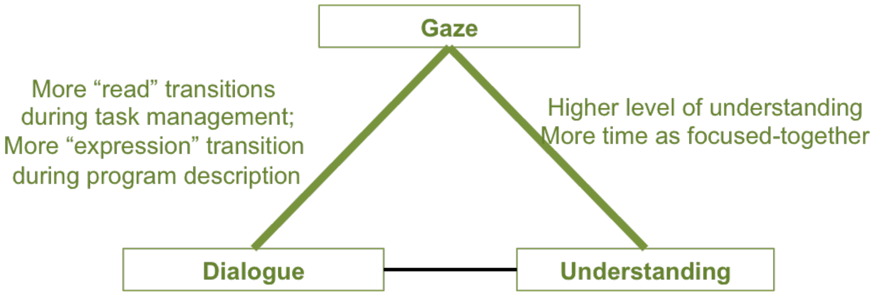


Figure 3.20 – *Contribution to the triumvirate relationship between the gaze, the dialogues and the level of understanding (figure 3.1) after combining all the three variables.*

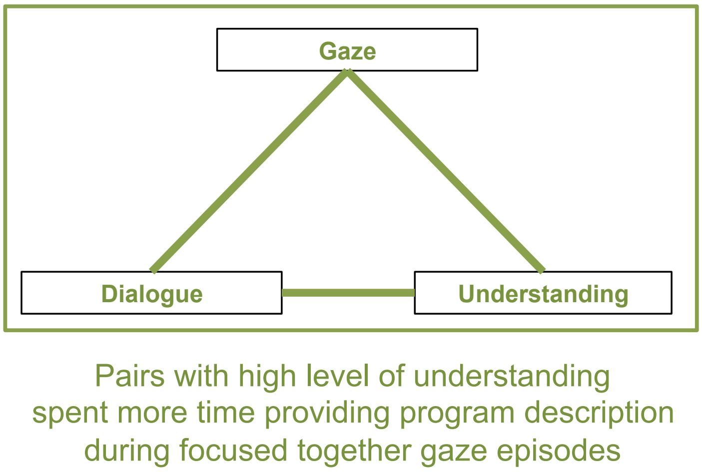


Figure 3.21 – *Contribution to the triumvirate relationship between the gaze, the dialogues and the level of understanding (figure 3.1) after combining all the three variables.*

and expressions, a transition type that reflected a data flow driven reading of the program. Conversely, pairs with a who got a sense of what the program is doing but were not able to provide the exact explanation, spent relatively more time parsing the program by systematically looking at all types of semantic elements. These findings were compatible with the findings from Jermann and Nüssli [2012] who found that for individual programmers, experts looked less than novices at structural elements (type names and keywords) which were not essential when understanding the functionality of the code. Experts looked more than novices at the predicates of conditional statements and the expressions (e.g. v /= 10;), which contain the gist of the programs. Our current findings confirmed these findings in the context of pairs by using an analysis of gaze transitions between semantic elements . Pairs with high level of understanding put relatively more individual efforts on understanding the entities and their relationships (data flow).

A possible explanation for this difference could be that for the pairs with low level of understanding some structural elements could act as “while demons” [Bonar and Soloway, 1983]. On other hand, pairs with high level of understanding showed “as-needed” strategy for building their understanding of the program based on their understanding of the relation between variables in the program [Koenemann and Robertson, 1991].

Moreover, in convergent fixation episodes, pairs with high level of understanding as well as pairs with low level of understanding tried to understand the program via a strategy of linear reading. This was depicted by their transitions between expressions and structural elements of the program. In comparison, the data flow transitions were less frequent in divergent fixation episodes for pairs in both the levels of understanding. A possible explanation for the differences between convergent and divergent episodes could be that, programmers were visually searching the code for variable and method names during the divergent phases and that in this case the augmentation of data flow transitions stemmed from a selective exploration of the code. Another explanation could be that, during divergent episodes, programmers focused on building basic knowledge about variables and expression which was then discussed during convergent episodes, where structural elements of the code were used to define the joint focus of attention. An analysis of the dialogue between partners would help to understand these subtle differences.

Our second perspective was concerned with the relation between gaze and dialogues. We found that while giving a high abstraction explanation, the participants were looking at different method definitions and while giving a low abstraction explanation the participants were looking at the system_methods. As we mentioned earlier, most system method calls were used for the interface messages. Guessing the program functionality from the interface messages was considered as low level abstraction. On the other hand, having a complete picture of the functionalities of different methods and the over all data flow (build upon the method calls) raised the level of abstraction in the program description.

The third perspective combined the gaze, dialogue and understanding at different temporal granularities (figure 3.13). This was an effort to present the interaction between the two programmers as a sequence of actions at different time scales and the main challenge was to bridge the gaps between the two consecutive time scales.

Concerning the bridge between two neighbouring time scales, we analysed each pair of time scales. We observed that the pairs with high level of understanding spent more time being "focused together" and while they are "focused together" the participants in the pair explained the functionality of the program to each other. When the pairs with high level of understanding were "not focused together" they talked about their next steps in the task (e.g., they talked about where to look next). On the other hand, pairs with low level of understanding exhibited the opposite behaviour as they spent more time being "not focused together".

Moreover, while the pairs with low level of understanding were "focused together" they talked about managing their focus and when they were "not focused together" the participants explained to each other a small part of the functionality of program to maintain a shared focus. Based on our observations, we think that this reflected different ways to understand the program. The "focused" way consisted of explaining in depth the functionality of the program, whereas the “unfocused” way consisted of describing the code to the partner and to “traverse” the code together.

One important observation was the interaction effect of the “level of understanding” and the “focus-similarity episodes” on the type of dialogues. There was no global relation between the gaze episodes and the dialogue episodes. However, we observed a direct relation between gaze indicators at the level of operations and dialogues. Irrespective of the level of understanding, the pairs had a higher proportion of “expressions” gaze transitions during “description” episodes. Moreover, the pairs had a higher proportion of “read” gaze transitions during “management” episodes. A possible explanation to this observation could be that, during a “description” episode the participants were more concerned with “what the program does?” This piece of information was contained in expressions of the program and hence the participants spent their time on understanding the expressions. On the other hand, during a “management” episode participants were talking about where to go next, or they were searching a particular piece of code; hence, the gaze of participants was as if they were scanning the code like an English text.

In this chapter, we presented the different (automatic and manual) ways to find the interaction episodes and to find the relationships between behaviour during different interaction episodes and comprehension. We found that the pairs with good understanding followed the data-flow in the program; while others (pairs with poor understanding), read the program as if they were reading a text. We also found a very close coupling between the gaze on different areas of the program and the level of abstraction in the dialogue. We found that gaze on the print messages was often accompanied by a low level of abstraction; and gaze on the key methods of the program was accompanied by a high level of abstraction in dialogue. Finally, we found that, during an episode of small focus size the pairs with good understanding talked about the functionality of the program while the others talk about task management. These results show that there is a triumvirate structure of relation between the cognition, gaze and dialogues. In the next chapters, we will explore this structure in a different context where we consider a special case of dyadic interaction as a teacher-student pair.

4 How Students Learn with MOOCs: An Exploratory Study

4.1 Introduction

In the last several years, millions of students worldwide have signed up for massive open online courses (MOOCs). The major issues we addressed are: how to make the learning process more efficient; and how to develop efficient means of capturing the attention and engagement of students. In this chapter, we present an exploratory eye-tracking study to shed some light upon “how to capture the attention of MOOC students?” This study was constrained in terms of it’s ecological validity (for example, the students were not provided with any control over the video playback and the slides were mostly textual) because our main focus was to ensure good data quality to be able to develop methods to highlight the differences among students based on their learning outcome. Moreover, in this study we did not consider the student as a single entity; but we analyse the interaction of the teacher-student dyad.

In this chapter, we start by laying out the context, i.e., massive open online courses (MOOCs). Then we provide the details of the experiment and different variables we used to analyse the interaction of the teacher-student pair. Finally, we present the results of the study and discussion. For this chapter our domain of investigation remains the the same as the previous chapter: the relation between cognition, communication and attention (measured using the students’ gaze). Instead of studying the cognition underlying program understanding, we study the cognitive processes responsible for learning; and instead of studying the communication between a pair of collaborators, we study a special case of a dyad, i.e., of teacher and student.

4.2 Context: Massive Open Online Courses

Massive open online courses (MOOCs) are online learning resources designed with intentions of reaching a large number of student population. The student population has no restrictions over age, ethnicity, area of expertise, employment status, job description or university degree. In other words, MOOCs are prepared for anyone who wants to take the course. The unbounded nature of MOOCs attracts a vast number of people from diverse backgrounds and expertise.

There are different ideologies driving content creation in MOOCs: *1)* cMOOCs or connectivist MOOCs are based on informal learning networks; and *2)* xMOOCs or content-based MOOCs are based on behavioural learning theories.

The key features of MOOCs and the differences from the traditional distance education are in the acronym used here. *1)* massive unlimited number of participants as opposed to relatively smaller number in distance learning. *2)* the courses are designed to be open to global audience, with no to a few prerequisites for participants and there is no participation fees. *3)* the courses are designed to be conducted strictly online and location-independent.

The unlimited number and the global nature of the students in a MOOC makes it very difficult to find successful learning processes among them. We focused on capturing the attention of the students while they attended the video lectures; and on finding the gaze patterns that were indicative of their success in achieving the learning outcome.

4.3 Problématique

This eye-tracking study is contextualised within a MOOC. We chose MOOC videos as stimulus for the eye-tracking because the effectiveness of video as a medium for delivery of educational content had already been studied and established in literature. In this chapter, we proposed to use the gaze-based variables, which were context free (did not require to define areas of interest on the stimulus), to differentiate between the levels of learning outcome. The benefit of using stimulus based variables was that these variables were generic enough to be computed for any kind of stimulus. Moreover, relation between performance and other behavioural constructs (for example, learning strategy) with such variables could be explained according to the stimulus type. The MOOC videos are usually diverse as per the content of the video is considered. Using stimuli-based variables in the analysis, might enable the researchers to analyse diverse content of the MOOC videos in a similar manner.

Another method we proposed in this chapter is to capture students' attention as a response to what the teacher was saying. We tackled this situation from the teacher's perspective: *"How much the student is with me?"* Accordingly, we called this gaze-measure **with-me-ness**: was the student really "following" the lecturer, i.e. paying attention to the elements of the display that correspond to the instant behaviour of the teacher? We selected two aspects of teacher's behaviour that could have influenced the students' attention: the teacher's dialogue and the teachers' deictic references. This study addressed the following methodological questions:

1) What are the gaze based variables that can be computed for a variety of stimulus and can be related to the performance and behavioural indicators?
2) How can we define attention through a gaze-measure? At what levels can we define the attention or from a teacher's perspective the measure of "With-me-ness"?

Apart from the methodological questions, in this chapter we addressed the following educational questions:

1) How are the gaze-based variables related to learning outcomes of students?
2) How are perceptual and conceptual levels of with-me-ness is related to learning outcomes of students?

4.4 Experiment

4.4.1 Participants and procedures

In the experiment, the participants watched two MOOC videos from the course "Functional Programming Principles in Scala" and answered programming questions after each video. Participants' gaze was recorded, using SMI RED 250 eye-trackers, while they were watching the videos. Participants were not given controls over the video for two reasons. First, the eye-tracking stimulus for every participant was the same which in turn facilitated the same kind of analysis for each of the participants. Second, the "time on task" remained the same for each participant.

40 university students from École Polytechnique Fédérale de Lausanne, Switzerland participated in the experiment. The only criterion for selecting the participant was that each participant took the Java course in the previous semester. Upon their arrival in the experiment site the participants signed a consent form, then they answered three self-report questionnaires for a 20-item study processes questionnaire [Biggs et al., 2001], 10-item openness scale and 10-item conscientiousness scale [Goldberg, 1999]. Then they took a programming pretest in Java (Appendix B). In the last phase of the experiment, they watched two videos from the MOOC course [1] and after each video they answered programming questions based on what they were taught in the videos (Appendix C).

4.4.2 Participant categorisation

Expertise: We used median split on the pretest score ($max = 9, min = 2, median = 6$) and we divide the participants in "experts" (more than or equal to the median score) and "novices" (less than the median score). The maximum and minimum scores for the pretest were 10 and 0, respectively.

Performance: We used median split on the posttest score ($max = 10, min = 4, median = 8$) and we divide the participants in "good-performers" (more than or equal to the median score) and "poor-performers" (less than the median score).The maximum and minimum scores for the posttest were 10 and 0, respectively.

Learning Strategy: We used median split on the study process questionnaire score ($max =$

[1]The MOOC "Functional Programming Principles in Scala" was given by Prof. Martin Odersky. This course was developed at École Polytechnique Fédérale de Lausanne, Switzerland.

42, $min = 16$, $median = 31.5$) and we divide the participants in “deep-learners” (more than or equal to the median score) and “shallow-learners” (less than the median score). The maximum and minimum for the study process questionnaire score were 20 and −20, respectively. For more details on the scoring procedure, see Biggs et al. [2001].

4.5 Process Variables

4.5.1 Content coverage

Heat-map variables: Attention points

Attention points are computed using the heat-maps (for details on heat-maps see Holmqvist et al. [2011]) of the participants. We divided the MOOC lecture in slices of 10 seconds each and computed the heat-maps for each participant. Following are the steps to compute attention points from the heat-maps:

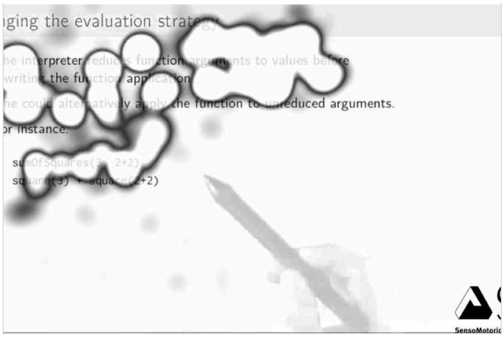
1) Subtract the image without heat-map (figure 4.1b) from the image that has the slide overlaid with heat-map (figure 4.1a).
2) Apply connected components on the resulting image (figure 4.1c)
3) The resulting image with connected components identified (figure 4.1d) gives the attention points.

Attention points typically represented the different areas where the students focused their attention. The number of the attention points would depict the number of attention zones and the area of the attention points would depict the total time spent on a particular zone. We compared the number of attention points and the average area covered by attention points per 10 seconds across the levels of performance and learning strategy. The area covered by the attention points typically indicated the content coverage for students. The content coverage indicates the content read by the students and the time spent on the content.

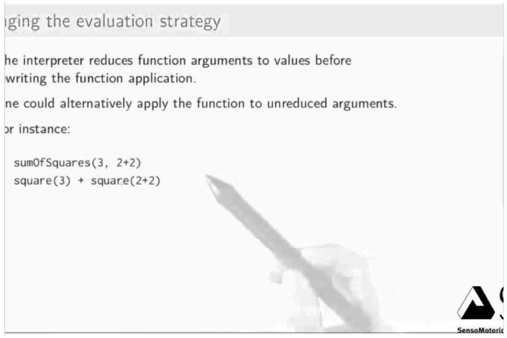
Scanpath variables

We computed two variables from students’ scan-paths. The number of areas of interest (AOIs) missed by the students and the number of AOIs re-watched by the students. Figure 4.2 shows a typical example of how these variables were computed.

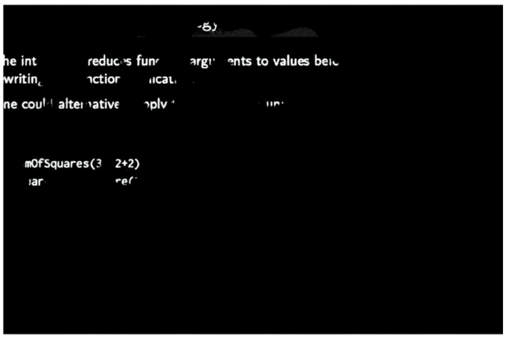
AOI misses: An area of interest (AOI) was said to be missed by a participant who did not look at that particular AOI at all during the period the AOI was present on the screen. In terms of learning behaviour AOI misses would translate to completely ignoring some parts of the slides. We counted the number of such AOIs per slide in the MOOC video as a scan-path variable and compare the number of misses per slide across the levels of performance and learning strategy (for details on areas of interest see Holmqvist et al. [2011]).



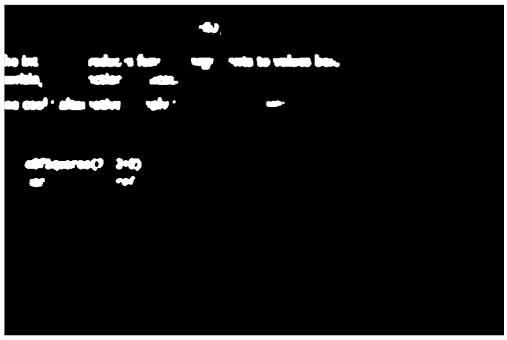
(a) *A slide with the overlay of 10 seconds' heat-map.*



(b) *A slide (same as figure 4.1a) without the overlay of heat-map.*



(c) *Resulting image after subtracting image without the heat-map (figure 4.1b) from heat-map overlaid image (figure 4.1a).*



(d) *Applying connected component on the figure 4.1c gives us attention points.*

Figure 4.1 – *Method to get the attention points and the area of the attention points.*

AOI backtracks: A back-track was defined as a saccade that went to the AOI which is not in the usual forward reading direction and had already been visited by the student. For example, in the figure 4.3, if a saccade goes from AOI3 to AOI2 it would be counted as a back-track. AOI back-tracks would represent rereading behaviour while learning from the MOOC video. The notion of term rereading in the present study was slightly different than what is used in existing research (for example, Millis and King [2001], Dowhower [1987] and Paris and Jacobs [1984]). The difference comes from the fact that in the present study the students did not reread the slides completely but they can refer to the previously seen content on the slide until the slide was visible. We counted the number of back-tracks per slide in the MOOC video as a scan-path variable and compared the number of back-tracks per slide across the levels of performance and learning strategy.

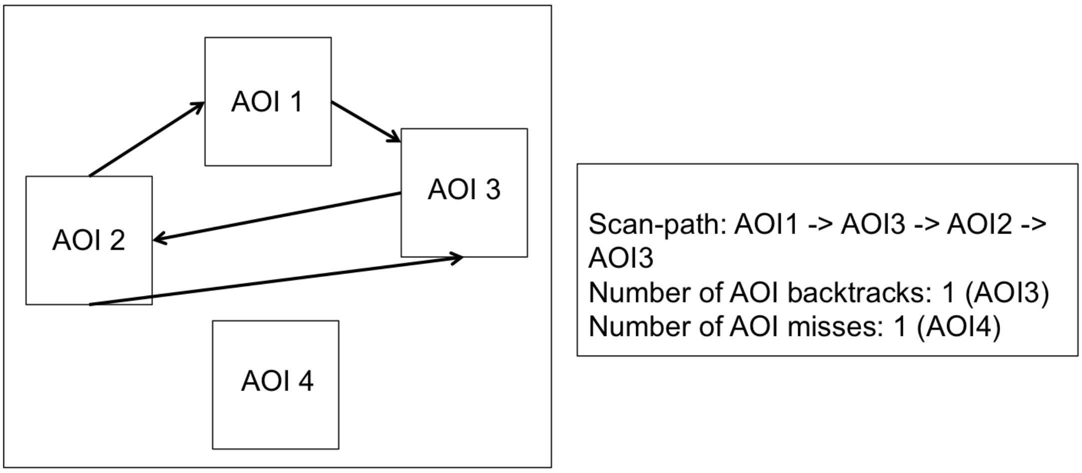


Figure 4.2 – *A typical example of a scanpath (left); and the computation of different variables (right).*

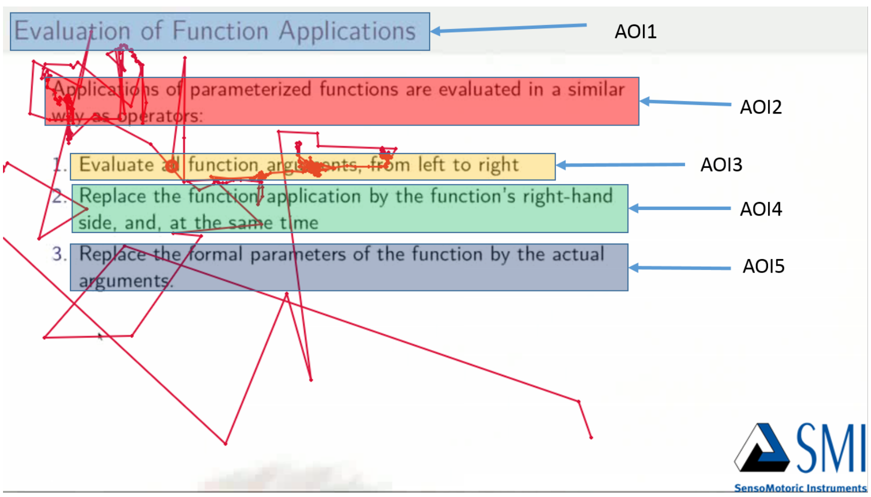


Figure 4.3 – *Example of a scan-path and Areas of Interest (AOI) definition. The rectangles show the AOIs defined for the displayed slide in the MOOC video and the red curve shows the visual path for 2.5 seconds.*

4.5.2 With-me-ness

With-me-ness is defined at two levels: perceptual and conceptual. There are two ways a teacher may refer to an object: with deictic gestures, generally accompanied by words ("here", "this variable") or only by verbal references ("the counter", "the sum"). Deictic references were recorded using two cameras during MOOC recording: first, that captured the teacher's face; and second, above the writing surface, that captured the hand movements. In some MOOCs, the hand is not visible but teacher used a digital pen whose traces on the display (underlining a word, circling an object, adding an arrow) act as a deictic gestures. Perceptual with-me-ness measured if the students looked at the items referred to by the teacher through deictic acts. Conceptual with-me-ness was defined using the discourse of the teacher: did students look at the object that the teacher was verbally referring to, i.e., that the teacher was referring to a set of objects that were logically or semantically related to the concept he was teaching. Figure 4.6 shows the relative temporal granularities of the two levels of with-me-ness and different levels of perceptual with-me-ness.

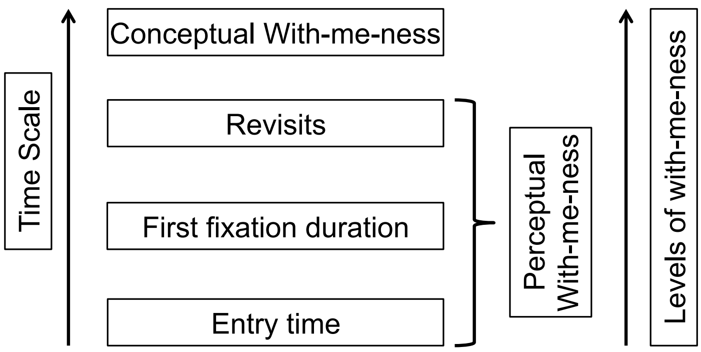


Figure 4.4 – *Temporal description of the two levels of with-me-ness and the sub-levels of perceptual with-me-ness.*

The notion of with-me-ness is also comparable with measures of gaze coupling that were developed in studies involving dual eye-tracking. Cross-recurrence [Richardson et al., 2007] reflected how much the gazes of two people followed each other during the interaction. Cross-recurrence was highest during references and cross-recurrence level was related to the quality of interaction [Jermann and Nüssli, 2012]. With-me-ness is defined at two levels:

Perceptual With-me-ness: The perceptual "with-me-ness" has 3 main components: entry time, first fixation duration and the number of revisits. *1)* Entry time was the temporal lag between the times a referring pointer appeared on the screen and stops at the referred site (x,y) and the time student first looked at (x,y). *2)* First fixation duration was how long the student gaze stopped at the referred site for the first time. *3)* Revisits were the number of times the student's gaze came back to the referred site.

Conceptual With-me-ness: The teacher may also verbally refer to the different objects on the slide. We measured how often a student looked at the object (or the set of objects) verbally referred to by the teacher during the whole course of time (the complete video duration). In order to have a consistent measure of conceptual "with-me-ness" we normalised the time a student looked at the overlapping content (the verbal reference and the slide content) by slide duration.

4.6 Results

4.6.1 General statistics

We observed no clear relation between the three variables (expertise, learning strategy and performance). There was no significant relation between expertise and performance ($\chi^2(df = 1) = 9.72, p > .05$). There was no significant relation between expertise and learning strategy ($\chi^2(df = 1) = 3.12, p > .05$). There was no significant relation between learning strategy and performance ($\chi^2(df = 1) = 4.18, p > .05$). Moreover, we did not observe any relationship between the gaze variables and the personality factor or the learning strategy.

4.6.2 Content coverage

Expertise vs. scan-path variables and attention points. We did not observe any significant relation between expertise and scan-path variable and attention points. Expertise had no relation with the number ($F(1,38) = 1.00, p > .05$) or the average area ($F(1,38) = 1.17, p > .05$) of the attention points. Moreover, expertise had no relation with AOI misses ($F(1,38) = 2.06, p > .05$) or AOI back-tracks ($F(1,38) = 4.00, p > .05$) of the attention points. In the following subsections, we report the relationships for the heat-map and scan-path variables with learning strategy and/or performance.

AOI misses and AOI-backtracks vs. Learning Strategy. There was no significant relation between the learning strategy and the number of area of interest (AOI) misses ($F(1,38) = 0.04, p > .05$) as well as the number of AOI back-tracks ($F(1,38) = 0.21, p > .05$).

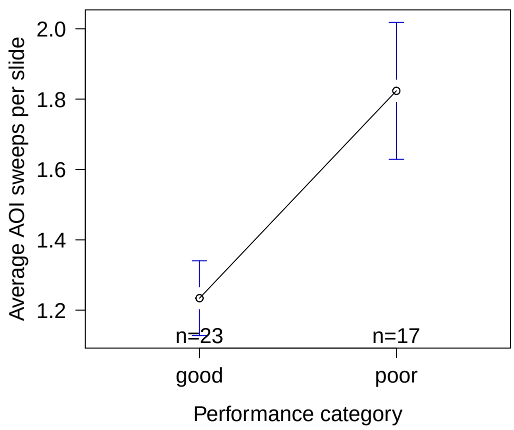
AOI misses and AOI-backtracks vs. Performance. The poor-performers missed significantly more AOIs per slide than the good-performers ($F(1,38) = 35.61, p < .01$, figure 4.5a). Whereas, the **good-performers back-tracked to significantly more AOIs per slide than the poor-performers** ($F(1,38) = 44.29, p < .01$, figure 4.5b). This suggested that the good-performers missed less content on the slide and reread more content than the poor-performers. We looked at the AOI misses every slide of the MOOC lecture and used a median cut on the number of AOI misses per student. We divided the AOI misses in high-misses and low-misses and compared the AOI misses across the performance levels. We observed that 65% of the poor-performers had low misses as compared to 87% of the good-performers ($\chi^2(df = 1) = 28.9, p < .05$).

Attention Points vs. Performance and Learning Strategy. We did not observe a difference in the number of attention points for good and bad performers ($F(1,38) = 1.00, p > .05$). Moreover, **there was no difference in the number of attention points for deep and shallow**

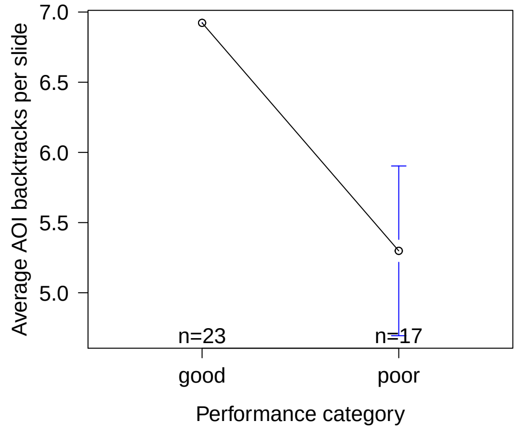
learners ($F(1,38) = 1.00$, $p > .05$). However, **the good-performers had significantly broader average area for the attention points than the poor-performers** ($F(1,38) = 5.47$, $p < .05$, figure 4.5c). Furthermore, **the deep-learners had significantly broader average area for the attention points than the shallow-learners** ($F(1,38) = 4.21$, $p < .05$, figure 4.5d). This suggested that, the good-performers spent more time reading the content than the poor-performers and the deep-learners spent more time reading the content than the shallow-learners. To confirm this we also measured the average reading time across the learning strategies and the levels of performance. A 2-way ANOVA shows two single effects. First, the **good-performers had a significantly higher average reading time than the poor-performers** ($F(1,36) = 9.99$, $p < .01$, figure 4.5e). Second, **the deep-learners had a significantly higher average reading time than the shallow-learners** ($F(1,36) = 4.26$, $p < .05$, figure 4.5f).

Table 4.1 – *Means and standard deviations for the different variables used in section 4.6.2 for learning strategy and performance categories.*

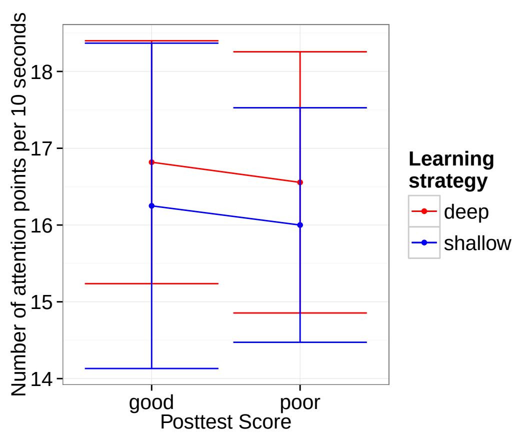
	Dependent Variables							
	Learning Strategy				**Post test score**			
	Deep		**Shallow**		**Good**		**Poor**	
Process Variables	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.	Mean	Std. dev.
Number of attention points	16.70	2.58	16.15	3.15	16.52	3.22	16.29	2.37
Average area (pixels) of attention points	537.4	96.15	408.8	132.84	510.57	133.84	422.41	113.54
Reading time [milliseconds]	96.62	49.48	72.63	30.80	99.88	47.07	63.98	23.65
AOI sweeps	1.51	0.39	1.46	0.47	1.23	0.25	1.82	0.38
AOI backtracks	6.20	1.22	6.27	1.02	6.92	0	5.30	1.17



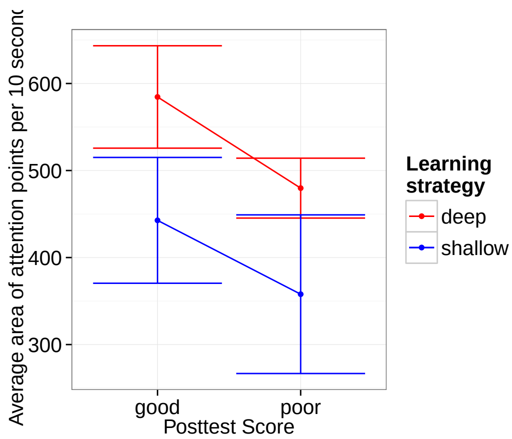
(a)



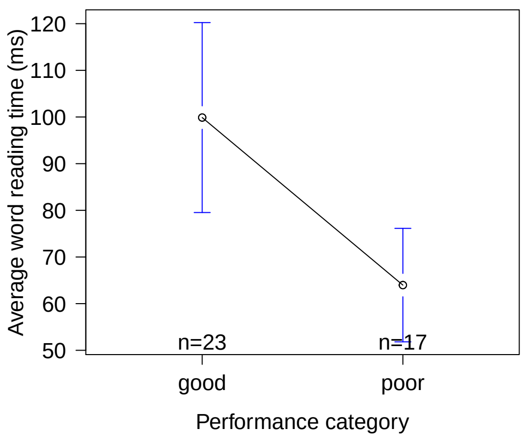
(b)



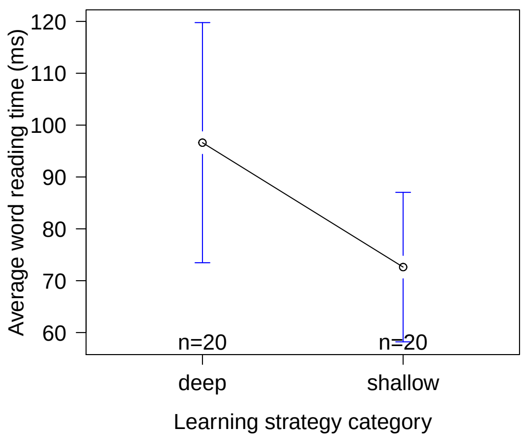
(c) The number of attention points.



(d) Average area of the attention points per 10 seconds.



(e) Reading time vs. performance.



(f) Reading time vs. learning strategy.

Figure 4.5 – *Mean plots and confidence intervals for attention point variables, scanpath variables and reading time across the different levels of learning strategy and performance.*

4.6.3 With-me-ness

Pretest score and with-me-ness: We did not observe any significant relation between pretest score and the two levels of with-me-ness.

Learning strategy and with-me-ness: We also did not observe any significant relation between learning strategy and the two levels of with-me-ness.

Posttest score and with-me-ness: We observed significant correlations for the two different levels of with-me-ness and the posttest score.

1) **Entry time**: We observed no correlation between entry time and the posttest score (Spearman's correlation = 0.1, $p > 0.5$, Figure 4.6a). This can be explained using the saliency of the teacher's pointer. When a moving object appears on the screen, it constituted a salient visual feature to which gaze was always attracted. This attraction did not reflect a deeper cognitive process and this is probably why it was not predictive of learning.

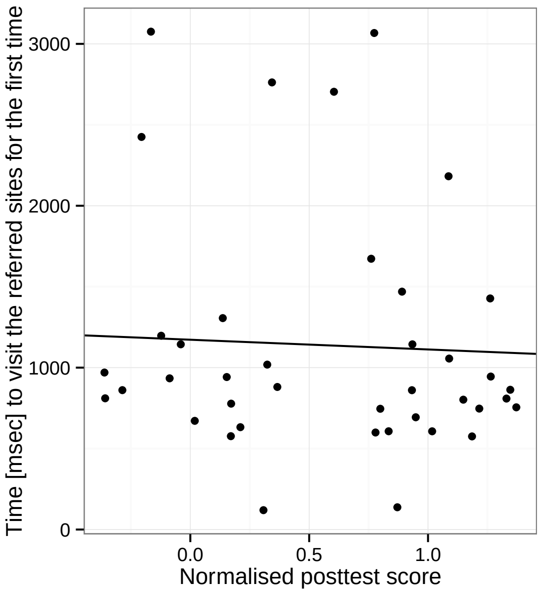
2) **First fixation duration**: We observed a significant correlation between the posttest score and the time spent for the first time the student looked at the referred site (Spearman's correlation= 0.35, $p < .05$, Figure 4.6b). **The students who scored high in the posttest were paying more attention to the teacher's pointers.** This behaviour is indicative of more attention during the moments of deictic references.

3) **Number of revisits**: We observed a significant correlation between the posttest score and the number of times the student looked at the referred site (Spearman's correlation= 0.31, $p < .05$, Figure 4.6c). **The students who scored high in the posttest came back to the referred sites more often than the students who scored less in the posttest.** Having more revisits also resulted in having more fixations and thus more aggregated fixation duration as well. The revisiting behaviour indicated rereading. Moreover, having more overall fixation duration on the referred sites indicated more reading time.

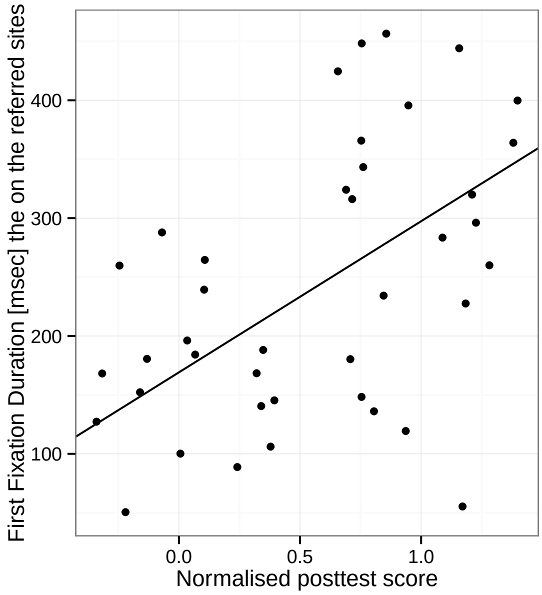
4) **Conceptual with-me-ness**: We observed a significant correlation between the posttest score and the time spent by the student following teachers' dialogues on the content of the slide (Spearman's correlation= 0.36, $p < .05$, Figure 4.6d). **The students who scored high in the posttest were paying more attention to the teacher's dialogue.** This behaviour was indicative of more attention during the whole video lecture.

4.7 Discussion

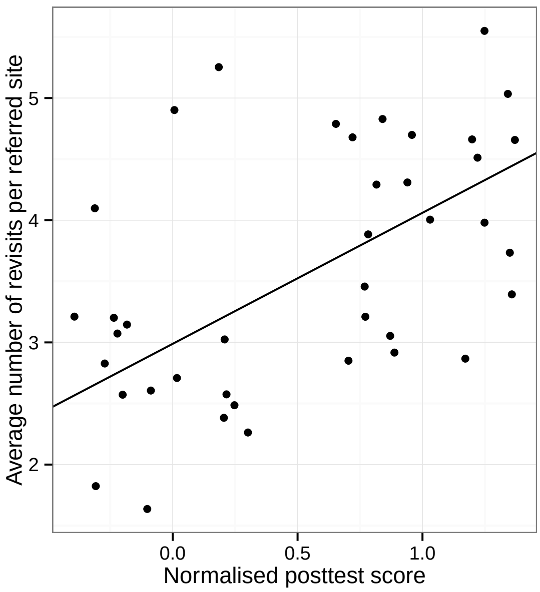
The attention points, derived from the heat-maps, were indicative of the students' attention both in the terms of screen space and time. The area of the attention points depended on the time spent on a specific area on the screen. Higher average area of the attention points could be interpreted as more reading time during a particular period. The good performing students having a deep learning strategy had the highest average area of the attention points per 10 seconds among all the participants, despite having the same number of attention points



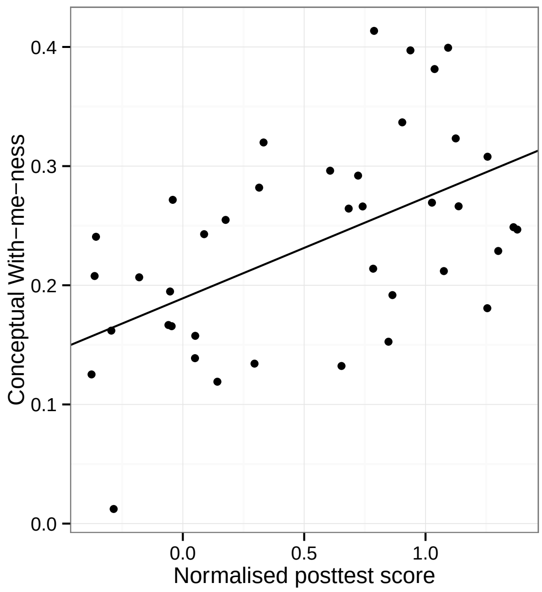
(a) *Entry time component of perceptual with-me-ness (x-axis) and posttest score (y-axis).*



(b) *First fixation duration component of perceptual with-me-ness (x-axis) and posttest score (y-axis).*



(c) *Revisits component of perceptual with-me-ness (x-axis) and posttest score (y-axis).*



(d) *Conceptual with-me-ness (x-axis) and posttest score (y-axis).*

Figure 4.6 – *Different with-me-ness components and posttest scores.*

during the same time period.

However, more reading time did not always guarantee higher performance. Byrne et al. [1992] showed the inverse in a longitudinal reading study by proving that the best performing students were the fastest readers. On the other hand, Reinking [1988] showed that there was no relation between the comprehension and reading time. As Just and Carpenter [1980] put "*There is no single mode of reading. Reading varies as a function of who is reading, what they are reading, and why they are reading it.*" The uncertainty of results about the relation between the performance and the reading time led us to find the relation between the reading time, performance and learning strategy. We found that the good-performers had more reading time than poor-performers and the deep-learners had more reading time than shallow-learners. We could interpret this reading behaviour, based upon the reading time differences, in terms of more attention being paid by the good performing students having a deep learning strategy than other student profiles. We could use attention points to give feedback to the students about their attention span. Moreover, one could use the attention points for student profiling as well based on the performance and the learning strategy.

The area of interest (AOI) misses and back-tracks were the temporal features computed from the temporal order of AOIs looked at. We found that good-performers had significantly fewer AOI misses than the poor-performers. AOI misses could be useful in providing students with the feedback about their viewing behaviour just by looking at what AOIs they missed.

The AOI back-tracks were indicative of the rereading behaviour of the students. We found that the good performers had significantly more back-tracks than the poor-performers. Moreover, the good-performers back-tracked to all the previously seen content, this explains the special distribution of AOI back-tracks for good-performers. Millis and King [2001] and Dowhower [1987] showed in their studies that rereading improved the comprehension. In the present study, the scenario is somewhat different than Millis and King [2001] and Dowhower [1987]. In the present study, the students did not read the study material again. Instead, the students referred back to the previously seen content again during the time the slide was visible to them. Thus the relation between rereading of the same content and the performance should be taken cautiously, clearly further experimentation is needed to reach a causal conclusion.

One interesting finding in the present study was the fact that the attention points had significant relationships with both the performance and the learning strategy. Whereas, the AOI misses and AOI back-tracks had significant relationships only with the performance. This could be interpreted in terms of the type of information we considered to compute the respective variables. For example, the attention-points' computation took into account both the screen space and the time information and AOI back-tracks (and misses) computation required only the temporal information. However, in the context of the present study, we could not conclude the separation between spatial and temporal information and how it effected the relation between the gaze variables and performance and learning strategy.

Next, we consider the results we got from with-me-ness. The entry-time component of the perceptual with-me-ness could be seen as the gaze behaviour when there was a salient element present on the visual stimulus [Parkhurst et al., 2002]. The pointer of the teacher appeared only a few times on the screen during the video lecture. We did not observed a correlation between the entry-time and the posttest scores. This could be explained by the fact that the pointer of teacher introduces a salient feature on the stimulus to which gaze is attracted. It did not reflect cognitive processing.

However, once the pointer was on the screen, the first fixation duration on the referred site was correlated with the posttest scores. The good-performers (those who scored high in the posttest) had more first fixation duration on the referred sites than the poor-performers. This was a typical situation during the moments of deictic references. Jermann and Nüssli [2012], in a pair-programming task, showed that better performing pairs had more recurrent gaze patterns during the moments of deictic references. Dale et al. [2011], in listening comprehension task, showed that the pairs having more recurrent gaze during the period of references performed better than the other pairs.

The revisit component of the perceptual with-me-ness can be seen as rereading behaviour. We observed a positive and significant correlation between the number of revisits to the referred sites and the posttest scores. The participants scoring high in the posttest had higher number of revisits to the referred sites. The explanation for this behaviour could be similar to the one with the AOI back-tracks.

The conceptual with-me-ness corresponded to a deeper form of attention, in terms of both the temporal scale and the cognitive effort "to be with the teacher". We observed a positive and significant correlation between the conceptual with-me-ness and the posttest scores. The conceptual with-me-ness can be explained as a gaze-measure for the efforts of the student to sustain common ground within the teacher-student dyad. Dillenbourg and Traum [2006] and Richardson et al. [2007] emphasised upon the importance of grounding gestures to sustain shared understanding in collaborative problem solving scenarios. A video was not a dialogue; the learner has to build common grounds, asymmetrically, with the teacher. The correlation we observed between conceptual with-me-ness and the posttest score seemed to support this hypothesis.

Finally, table 4.2 summarises the variables we introduced in this chapter. The comparison is based on two facts. First, is it possible to automatise the calculation of the variable; and second whether and how much pre-processing is required?

In a nutshell, we could say that the students who scored better in the posttest, had more content coverage and they were following the teacher, both in deictics and discourse, in an efficient manner than those who did not score well in the posttest. The results were not surprising, but could be utilised to inform the students about their attention levels during MOOC lectures. In the next chapter, we will see that the nature of the findings remains the same as we moved from a very controlled lab study to another lab study which was more ecologically valid.

Table 4.2 – *Comparison of different variables in terms of automatisation and pre-processing required.*

Measure Name	**Real-time computation**	**Pre-processing required**	**Type of pre-processing**
Heat-map variables	Yes	No	None
Scan-path variables	Yes	Yes	Defining the areas of interest (AOIS)
Perceptual with-me-ness	Yes	No	None
Conceptual with-me-ness	Yes	Yes	Transcribing the teacher's dialogues

5 Dual Eye-tracking Study in MOOC Context

5.1 Introduction

The study presented in this chapter answers a key question in eye-tracking research. In previous two chapters, we found two different results in two different settings of dyadic interaction. First, in a collaborative setting, we found that the collaborative performance was correlated to the amount of time the pair spent looking at the similar parts of a program (high gaze similarity). Second, in an individual eye-tracking study, we found that the posttest scores were correlated to the amount of time students spend in following the teacher's deixis and dialogues (high with-me-ness). In this chapter, we ask ourselves whether there exists relation between these individual and collaborative gaze patterns?

In order to answer this question,we designed an experiment which comprised of two tasks: an individual video lecture and a collaborative concept map task. The video lecture task also improved upon the study presented in the previous chapter, that had limitations in the terms of its ecological validity (no playback control, mostly textual slides). The participants had no control over the video playback. The reason for not giving them the video playback was to ensure an easy way to analyse the gaze data to compare the gaze-based variables against the learning outcome.

The main changes we introduced in the current study are: first, we gave the full video playback control to the participants. Second, we added an additional add-on collaborative activity for the participants. Finally, we introduced two different methods of priming [1] the students about the lecture content.

In this chapter, we first describe the concept of priming, as it had been used in this experiment. Second, we layout the research questions addressed in this study. Third, we give the details about the experiment. Fourth, we present the result and finally we give possible explanations

[1] The concept of "priming" used in this chapter is not the same as it had been used in classical psychology research. As we will introduce in the next section, we simply mean to introduce a few key elements from the lecture to the students, before they receive any learning material.

to the results. For this chapter, the conceptual domain remains the same as the previous chapter, i.e., the relation between the cognition, communication and attention. As in the previous chapter, we study the dyad of the teacher-student pair; and also dyads of collaborating students.

5.2 Activating student knowledge via priming

Priming or activating student knowledge (ASK), is giving a prior introduction to the lecture content to the students [Tormey and LeDuc, 2014]. Tormey and LeDuc [2014] conducted a study where they taught a content with which the students were completely unfamiliar. One half of the class got a small priming through a brainstorming session and the other half did not get any priming. The results showed that students who got priming had better learning gain than students who did not get any priming. The authors further mentioned -

"ASK" (Activating Student Knowledge) therefore, involves using questions during the introduction to a lecture to activate students prior knowledge related to the topic. This can be done out loud (using brainstorming) but can also be done as a quiet activity in which students respond in writing to a small number of targeted prompts." - [Tormey and LeDuc, 2014]

In this experiment, we took ASK to one step further. We used the pretest as a way to ask questions, but introduced a new version of priming via pretest, i.e., one half of the students got a simple textual pretest and the other half got the same pretest depicted as schemas.

5.3 Problématique

We conducted a dual eye-tracking study where the participants attended a MOOC lecture individually and then collaborated in pairs to create the concept map about the learning material. We used the pretest to shape the processing of the video content by the participants in a specific way (paying more attention to textual or schema elements in the video). We called this priming effect. This experiment was driven by two hypotheses.

The first hypothesis concerns the effect of priming on the gaze. There could be two possibilities: first, the **replication hypothesis**, the students would follow the similar elements as they were primed with (students in textual priming condition would concentrate more on the textual elements of the lecture). Second, the **compensation hypothesis**, the students would compensate for their method of priming (students in textual priming condition would focus more on the schema based elements in the lecture).

The second hypothesis was that there are two factors shaping the learning gain of the students: *1)* how closely students follow the teacher, *2)* how well they collaborate in the concept map task. The more a student follows the teacher, the more (s)he could learn (figure 5.1); the better a student collaborates with the partner, the more the pair could discuss the learning material and have a better understanding and hence achieve a better learning outcome.

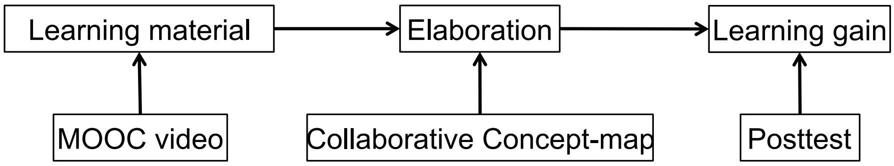


Figure 5.1 – *Schematic representation of the second hypothesis for the experiment. We hypothesise that the students would have higher learning gain provided, they follow the teacher in the video and they collaborate well with their partners during the concept map phase.*

Through this study we addressed the following research questions:

1) How does priming affect the gaze patterns (both in the individual and collaborative tasks) and the learning gain of the participants?

2) What is relation between the individual gaze patterns and the collaborative gaze patterns and how do these affect students' learning gains?

5.4 Experiment

5.4.1 Participants and procedure

There were 98 master students from École Polytechnique Fédérale de Lausanne participating in the present study. There were 20 females among the participants. The participants were compensated with an equivalent of CHF 30 for their participation in the study. There were 49 participants in each of the priming conditions (textual and schema). For the collaborative concept-map task, we had 3 pair configurations (based on their priming conditions): both the participants had textual priming (TT), both the participants had schema priming (SS), participants had different priming (ST). There were 16 pairs in each of TT and SS pair configurations while there were 17 pairs in ST pair configuration. The flow of experiment is shown in figure 5.2.

Upon their arrival in the laboratory, the participants signed a consent form. Then the participants took an individual pretest about the video content (Appendix D and E). Then the participants individually watched two videos about "resting membrane potential". Then they created a collaborative concept-map using IHMC CMap tools [2]. Finally, they took an individual posttest (Appendix F). The videos were taken from "Khan Academy" [3] [4]. The total length of the videos was 17 minutes and 5 seconds. One important point worth mentioning here is that the teacher was not physically present in the video.

The participants came to the laboratory in pairs. While watching the videos, the participants

[2] "CMap tools"
[3] "Resting Membrane Potential-Part 1"
[4] "Resting Membrane Potential-Part 2"

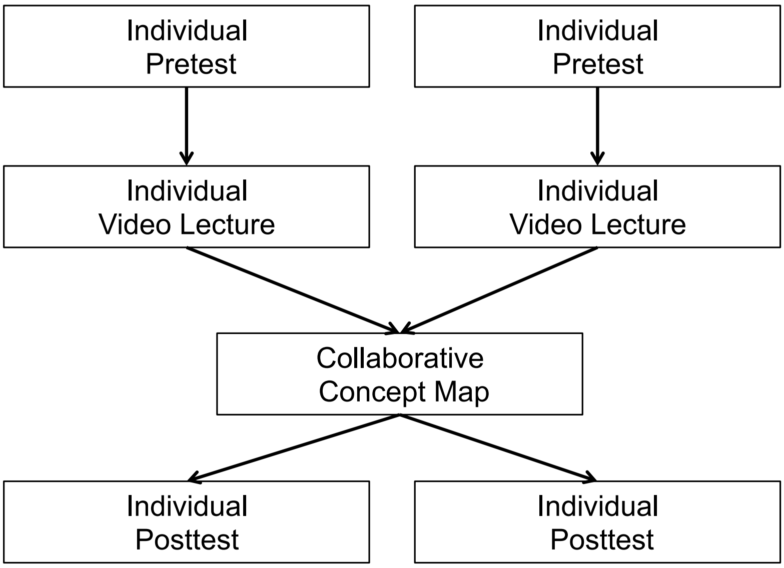


Figure 5.2 – *Schematic representation of the different phases of the experiment.*

had full control over the video player. The participants had no time constraint during the video-watching phase. The collaborative concept-map phase was 10-12 minutes long. During the collaborative concept-map phase the participants could talk to each other while their screens were synchronised, i.e., the participants in the pair were able to see their partners' actions. Both the pretest and the posttest were multiple-choice questions where the participants had to indicate whether a given statement was either true or false.

5.4.2 Independent variable: Priming

As we mentioned previously, we wanted to observe the difference in the gaze patterns for different modes of priming. We used a pretest as a priming method. We designed two versions of the pretest. The first version had textual questions (Appendix D). The second version had exactly the same questions as in the first version but they were depicted as a schema (Appendix E). Figure 5.3 shows one question from schema based pretest. The corresponding question in the textual pretest was: *"State whether the following statement is true or false: The main cause for the creation of resting membrane potential is more positive ions move inside the membrane than outside of the membrane."* Based on the two priming types, we had two priming conditions for the individual video lecture task: *1)* textual priming, and *2)* schema priming. The selection of the two priming methods (textual and schematic) was based on the fact that the MOOC videos are usually a mixture of the textual and schematic elements. We hypothesised that we could prime the students to look at either the textual or the schematic elements of the lecture. Hence, the the priming methods should have been consistent with the representation style of the MOOC lecture.

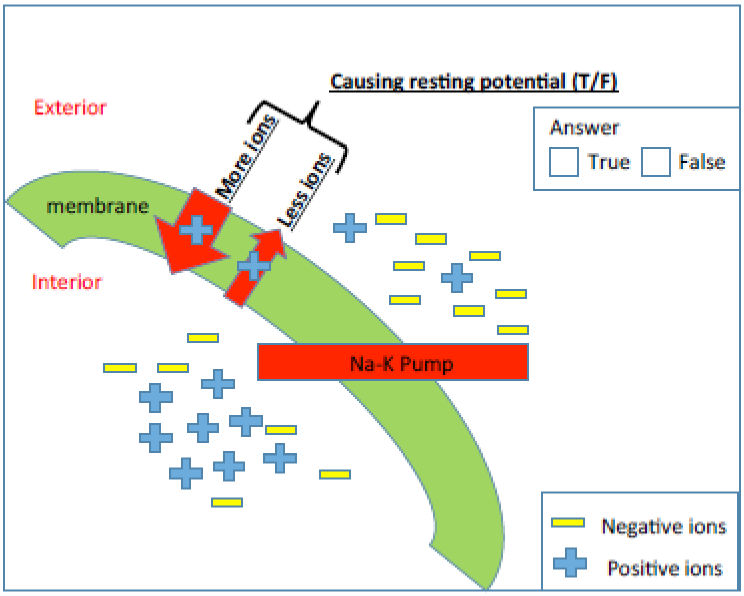


Figure 5.3 – *Example question from the schema version of the pretest. The corresponding textual question was "State whether the following statement is true or false: The main cause for the creation of resting membrane potential is more positive ions move inside the membrane than outside of the membrane."*

5.4.3 Independent variable: Pair configuration

Based on the two priming types we had three pair compositions for the collaborative concept map task: *1)* Both the participants received the textual pretest (TT); *2)* Both the participants received the schema pretest (SS); *3)* Both the participants received different pretests (ST).

5.4.4 Dependent variable: Learning gain

The learning gain was calculated simply as the difference between the individual pretest and posttest scores. The minimum and maximum for each test were 0 and 10, respectively.

5.4.5 Process variables

With-me-ness during individual video lecture task

As described in Chapter 4, with-me-ness was a gaze measure for quantifying students' attention during the video lectures. It has two components: *1)* perceptual with-me-ness and *2)* conceptual with-me-ness. The perceptual with-me-ness captured the students' attention especially during the moments when the teacher made explicit deictic gestures, whereas the conceptual with-me-ness captured whether and how much the gaze of the student was following the teacher's dialogues. To compute conceptual with-me-ness in this study, we mapped the teachers' dialogues to the different objects on the screen. We named them as objects of interest (Figure 5.4). Once we had the objects of interest on the screen, we computed what proportion of gaze time to the dialogue length (+2 seconds) in time is spent by the participants on the objects of interest. This proportion was the measure of the conceptual with-me-ness. There are a few moments where the explicit deictic gesture s accompanied by a verbal explanation, we consider these moments to be a part of time where we compute the perceptual with-me-ness. To compute the conceptual with-me-ness, we only consider those moments where there is only a verbal explanation to the lecture content on the screen.

Teacher: "so you have one force, the concentration driving K out; and another force the membrane potential, that gets created by its absence that's gonna drive it back in."

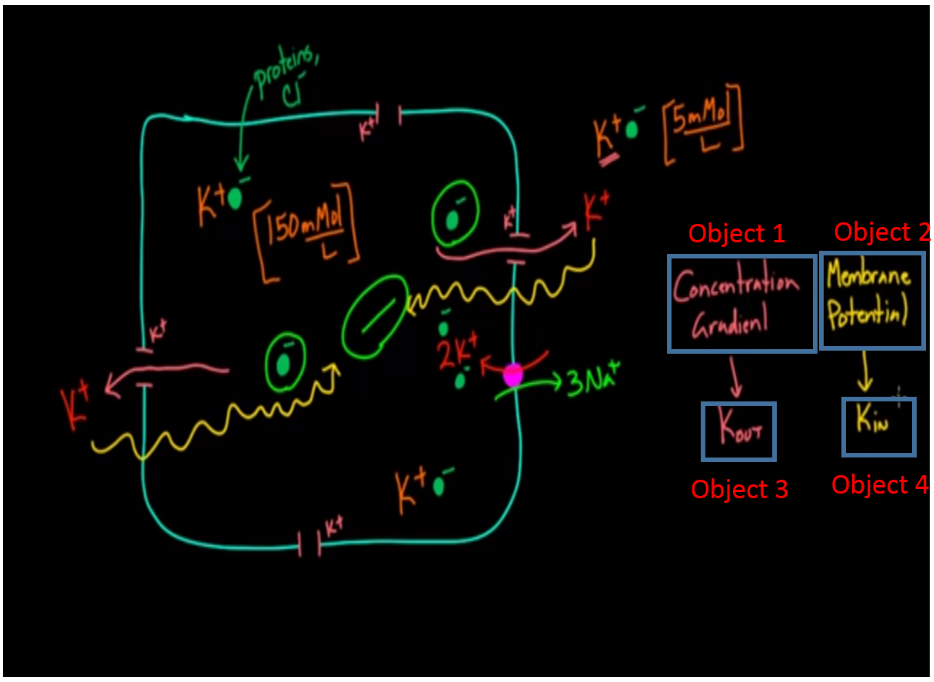


Figure 5.4 – *Example of areas of interest used in the experimental task. Objects 1 and 2 are textual elements, while object 3 and 4 are schema elements. The main schema in the middle of this snapshot was also divided into different schema elements like "ions", "membrane" and "channels".*

Gaze on textual elements during the individual video lecture task

The video lecture had a mix of textual and schema elements. The teacher drew some figures and charts during the lecture and also made some tables and wrote some formulae. We categorised the tables, formulae and the sentences written by the teacher as the textual elements of the video; and the graphs, figures and charts were categorised as schema elements. For example, figure 5.4 is a snapshot of the video we used in the experiment. The objects on the screen were divided into schema or textual objects of interest. We measured the time spent on the textual elements by the participants during the video lecture. This helped us verifying our hypotheses concerning the effect of priming (replication or compensation) on the gaze of the participants.

Gaze compensation during individual video lecture task

The proportion of time that the participants spent looking at the textual elements of the video did not correctly reflect the compensation in the gaze patterns, as the schema and textual elements did not appear in the same proportions on the screen throughout the video lecture. Initially, for a few minutes, the video contained only schema elements and later the teacher kept adding the textual elements. This made the proportions of schema and textual elements change over time. Hence, we needed to take this change into account to compute the real compensation effect. We proposed a gaze compensation index to be computed as follows:

$$\text{Gaze compensation index} = \sqrt{\sum \frac{\left(\frac{G_t}{G_s} - \frac{P_t}{P_s}\right)^2}{\frac{P_t}{P_s}}}$$

Where,
G_t := Gaze on textual elements in a given time window;
G_s := Gaze on schema elements in a given time window;
P_t := Percentage of screen covered with textual elements;
P_s := Percentage of screen covered with schema elements;

A gaze compensation index equal to zero reflects that the participant spent the same proportion of time on textual and schema elements, as they were present on the computer screen. On the other hand, a higher gaze compensation index indicated a higher difference between the proportion of time spent on the textual and schema elements than the proportions of screen space they covered.

Gaze similarity during collaborative concept map task

The gaze similarity during collaborative concept map task was calculated using the same method as described in Chapter 3.

5.5 Results

The order of the results will follow the same structure as the description of the variables. First we show the results concerning the effect of priming on different gaze variables we proposed. Second, we present results showing the relation of individual and collaborative gaze measures the learning gain.

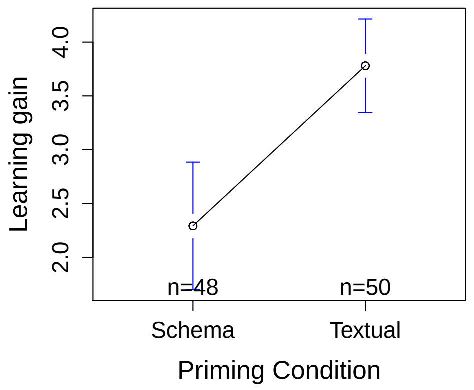
5.5.1 Effect of priming

1) **Learning gain:** An ANOVA with prior knowledge activation methods as a between subject factor showed a significant difference in the learning gain between the two priming conditions (figure 5.5a). **The learning gain for the participants in the textual priming condition was significantly higher than the learning gain for the participants in the schema priming condition** ($F[1,96] = 16.77, p < .01$). Furthermore, an ANOVA with pair composition as a between subject factor showed a significant difference in the learning gain between the three pair compositions (TT, TS, and SS). **The learning gain for the TT pairs was the highest and the learning gain for the SS pairs was the lowest** ($F[2,46] = 6.18, p < .05$).

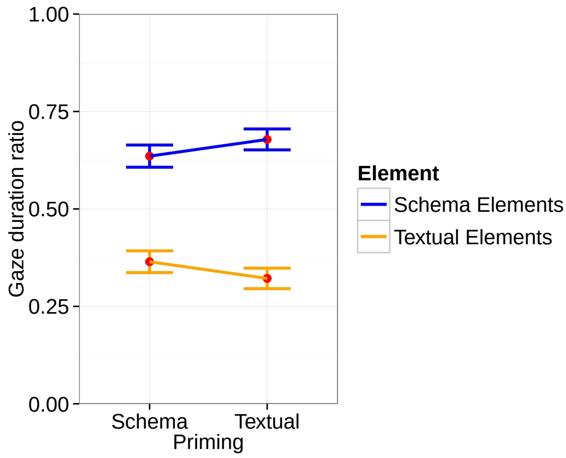
2) **Time on text:** An ANOVA with prior knowledge activation methods as a between subject factor showed a significant difference in the time spent on the textual object in the video between the two priming conditions (figure 5.5b). **The time spent on video for the participants in the textual priming condition was significantly lower than the learning gain for the participants in the schema priming condition** ($F[1,96] = 4.49, p < .05$).

3) **Gaze compensation:** An ANOVA with prior knowledge activation methods as a between subject factor showed the gaze compensation index across the two priming conditions (figure 5.5d). **The participants in the textual priming condition had higher compensation index than the participants in the schema priming condition** ($F[1,96] = 56.198, p < .001$).

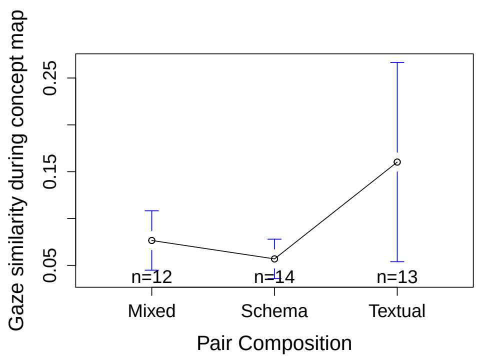
4) **Gaze Similarity:** An ANOVA with pair composition (TT, TS, and SS) as a between subject factor showed a significant difference in the gaze similarity between the three pair configurations (figure 5.5c). **The gaze similarity for the TT pairs was significantly higher than the gaze similarity for ST and SS pairs** ($F[1,37] = 3.77, p < .05$). The levels of gaze similarities were very low (the scale being 0 to 1). However, the baseline was the probability of two people looking at the same time at one of 14 objects on the screen, i.e., $1/2^{14}$.



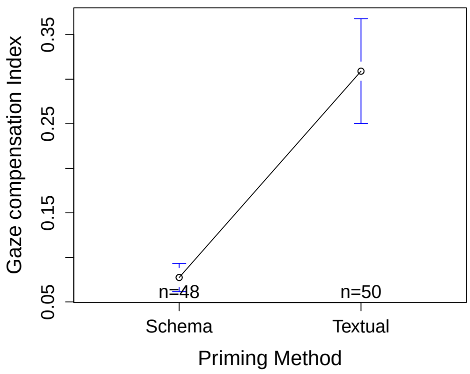
(a) *The mean plots for learning gain across two priming conditions.*



(b) *Mean plots for gaze proportions on textual and schema based elements for two priming conditions.*



(c) *Gaze similarity for pairs in three different pair compositions.*



(d) *The mean plots for compensation across two priming conditions.*

Figure 5.5

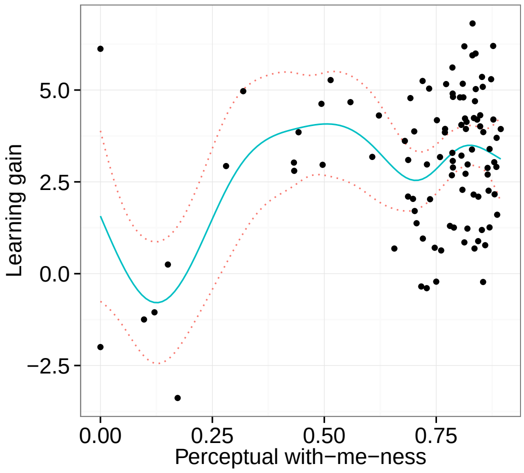
5.5.2 Individual with-me-ness, collaborative gaze similarity and learning gains

We present the results from the generalised additive models over the with-me-ness, gaze similarity and the learning gains. We observed, in a preliminary analysis, that the relations between these variables were non-linear. Hence, a linear correlation would not have worked in this case. We interpret the relation found between the three variables (with-me-ness, gaze similarity and the learning gains) as a non-linear correlation based on the value of R^2. This statistic tells us that how accurately we can predict the value of the second variable given the value os the first variable. To avoid the overfitting, in some cases, we divided the data into training and testing sets and checked whether the R^2 value were similar or not. We found similar R^2 values for both the training and testing sets for each of the following relations:

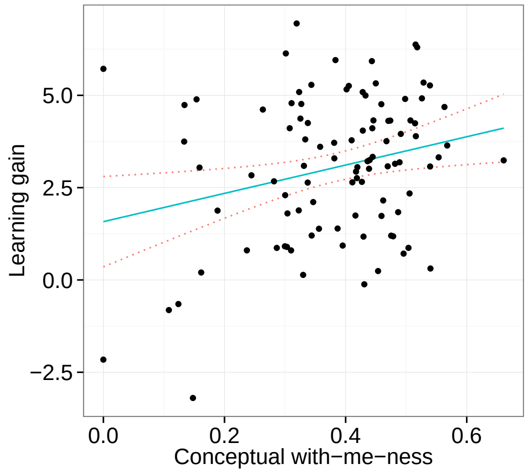
With-me-ness and learning gains: Both the components of with-me-ness were significantly correlated with the learning gain.We observed a significant positive correlation between the perceptual with-me-ness and the learning gain ($R^2 = 0.21, F(6.17, 7.30) = 3.85, p < .001$, figure 5.6a). This difference was irrespective of the priming condition. **The participants having high perceptual with-me- ness, had high learning gain**. We also observed a significant positive correlation between the conceptual with-me-ness and the learning gain ($R^2 = 0.06, F(1, 1) = 6.43, p < .05$, figure 5.6b). This difference was irrespective of the priming condition. **The participants having high conceptual with-me-ness, had high learning gain.**

With-me-ness and gaze similarity: We found the individual with-me-ness and collaborative gaze similarity to be positively correlated. We observed a significant positive correlation between the gaze similarity and the average perceptual with-me-ness of the pair ($R^2 = 0.98, F(8.22, 8.83) = 193.9, p < .001$, figure 5.6d). **The pairs having higher gaze similarity have higher average perceptual with-me-ness.** We also observed a significant positive correlation between the gaze similarity and the average conceptual with-me-ness of the pair ($R^2 = 0.58, F(2.93, 3.62) = 12.36, p < .001$, figure 5.6e). **The pairs having higher gaze similarity had higher average conceptual with-me-ness.**

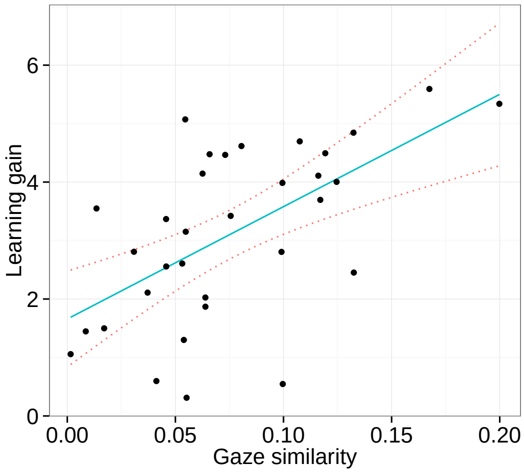
Gaze similarity and learning gains: We observed a significant positive correlation between the gaze similarity and the average learning gain of the pair ($R^2 = 0.34, F(1, 1) = 17.23, p < .001$, figure 5.6c). **The pairs having higher gaze similarity had higher average learning.**



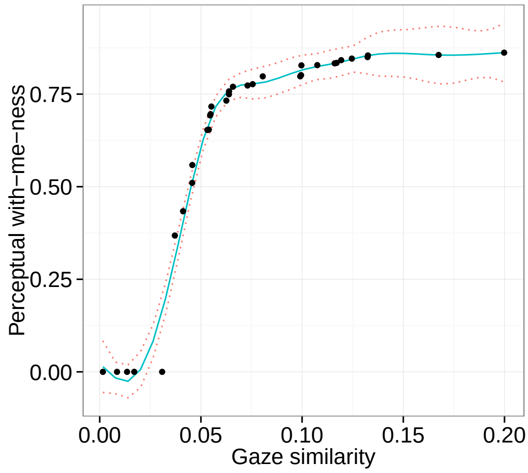
(a) Perceptual with-me-ness and learning gain



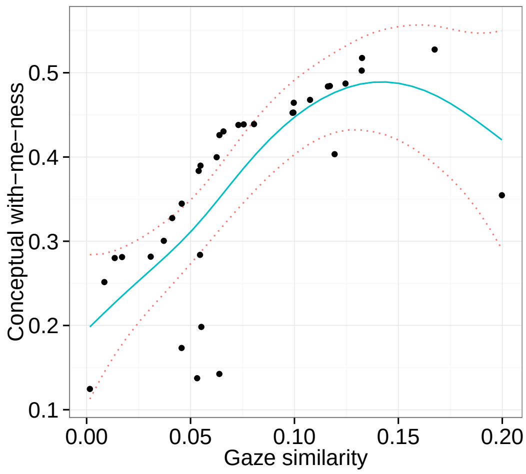
(b) Conceptual with-me-ness and learning gain



(c) *Gaze similarity (x-axis) and learning gain (y-axis).*



(d) *Perceptual with-me-ness during individual video watching(y-axis) and gaze similarity during collaborative concept-map task(x-axis).*



(e) *Conceptual with-me-ness during individual video watching (y-axis) and gaze similarity during collaborative concept-map task(x-axis).*

Figure 5.6

5.6 Discussion

The first question concerned the effectiveness of priming on the learning gain and gaze patterns (individual and collaborative) of the participants. The learning gain of the participants in textual priming condition was significantly higher than that for the participants in the schema priming condition (figures 5.5a). The explanation for this effect could be based on the theory of Tormey and LeDuc [2014]about Activating Student Knowledge (ASK) using priming methods. Tormey and LeDuc [2014] compared the students' learning gain with and without the priming in a history lecture. The priming method used in the study was a pretest. We extended the concept by using two different versions of pretest (textual and schema based). The textual method for ASK emerges as a better priming method than the schema method. A plausible reason for the effect on learning gain could be that the textual version gave more exact terms to look forward for in the lecture than the schema version of the pretest.

Moreover, we also found a relation between priming and the gaze during individual and collaborative tasks. We found that the participants in textual priming condition looked more at the schema elements of the video and the participants in schema priming condition looked more at the textual elements of the video (figure 5.5b). This was a compensation effect of the priming. This supported the compensation hypothesis from section 5.3. We also computed the gaze compensation effect based on the ratio of the textual and schema elements present on the screen and the ratio of the gaze on them respectively (figure 5.5b). The participants in the schema priming condition under-compensated for the priming they received in the video phase and hence they missed some of the key concepts. This could have a detrimental effect on their learning gains.

Furthermore, during the collaborative concept map task, the pairs with both the participants from the textual priming (TT) condition had higher gaze similarity than the pairs in other two configurations (ST and SS pairs). Once again, we could expect a better priming effect in textual priming condition than in the schema priming condition. The participants in the TT condition had better priming and they had better compensation for the key concepts from the lecture. This enabled them to elaborate together on the concepts in the collaborative concept map task and hence they had higher gaze similarity (figure 5.5c).

The second question we addressed, concerned the relations between the individual and collaborative gaze patterns, and students' learning gains. The two components of with-me-ness were positively correlated with the learning gain (figures 5.6a and 5.6b), which was consistent with the results found in the previous chapter. The only difference is that, in this study, we observed higher values for the perceptual and the conceptual with-me-ness than what we observed in the previous chapter. The different levels of with-me-ness values could be explained by the different types of the video lectures. The video used in chapter 4 had only textual slides. The video in the this experiment had no slides; the teacher started with a blank board and incrementally fills the board by writing the lecture material (schemas, tables, formulas). The higher values of the with-me-ness components in this experiment could be explained by the nature of the videos. In the video from chapter 4, the whole content is on the screen from the beginning of slide resulting in the distraction as students might start reading from the slides and do not listen to the teacher. On the other hand, the video content in the video of this experiment itself followed the flow of teacher's discourse and hence might have resulted in higher values of with-me-ness for every student.

Moreover, the pairs with high gaze similarity also had high average learning gain (figure 5.6c). A similar pair (in terms of gaze) elaborated on the lecture concepts in a better manner than the pair with low gaze similarity. More specifically, the pair with high gaze similarity worked on the same part of the concept map in a given time window, hence they developed a better shared understanding about the concerned topic. Whereas, the pair with low gaze similarity worked on less similar parts of the concept map and hence they failed to have a shared understanding.

Furthermore, the key question addressed in this chapter was about the relationship between the gaze patterns of the participants during the individual video watching phase and during the collaborative concept map phase (section 5.1). The pairs who had high average with-me-ness also had high gaze similarity (figures 5.6d and 5.6e). This could be explained in terms of sharing a strong basis for creating a shared understanding of the topic. If both of the participants followed the lecture in an efficient manner, i.e., with high with-me-ness, the pair had a strong base to build and maintain a shared understanding. Hence, the pair had more gaze similarity. This result was also consistent with the related research by Richardson et al. [2007] and Richardson and Dale [2005] where the gaze cross-recurrence is higher when the participants had a better level of shared understanding.

From the last three chapters, what had emerged is a concept of "looking through" versus "looking at": some learners look "at" the display, as we look at a magazine, while other students seem to look "through" the display, that is, to look at the teacher or their partner in interaction as if they were actually present there. The latter seems to gain deeper engagement and hence a better learning outcome. The students who looked "at" the display lag in following either the teacher or their partners, whereas the students who looked "through" the display, use the display not only to follow the teacher or their partner but they use the display to create a shared understanding. Having a shared understanding in turn increases the learning gain for such students.

The concepts of "looking through" and "looking at" could be seen as new interaction style categories. "Looking at" the interface/display indicates that the person is engaged with the material only, which is made available to him/her. "Looking through" the interface/ display indicates that the person is engaged with the peer. The peer in the video phase is the teacher and in the collaborative concept map is the collaborating partner. The "looking through" interaction resembles the social co-location of the interacting peers. As an analogy, to highlight the difference between the two interaction styles, we can compare the interaction with the teacher/collaborating partner to watching a movie. "Looking at" can be compared with liking the movie; whereas, "looking through" can be compared with appreciating the direction.

6 Gaze Aware Feedback: Effect on Gaze and Learning

6.1 Introduction

In chapters 4 and 5, we established the relation between students' gaze patterns and their learning outcome. We found in two different experiments that the students who followed the teacher's references and dialogues achieved higher learning results than those who did not. Students' with-me-ness levels were found to be correlated with their learning gains. In this chapter, we exhibit a method to improve students' attention levels, in other words their with-me-ness, by giving them feedback based on how well they follow the teacher in the video lecture. We present a study exploring the effects of gaze aware feedback during video lecture on students' with-me-ness and their learning gain.

6.2 Context

Gaze awareness had been used to build intelligent tutoring systems [D'Mello et al., 2012, Wang et al., 2006, Jaques et al., 2014], online collaboration support [Oh et al., 2002, Tan et al., 2009], query expansion systems [Buscher et al., 2008], and attention aware systems [Toet, 2006]. D'Mello et al. [2012] used students' real time gaze information to inform the tutor about the boredom and engagement levels for selecting the dialogue moves for the virtual tutor accordingly. The authors found that the gaze-aware tutor was more effective in terms of both maintaining a higher engagement level and achieving a higher learning gain. Wang et al. [2006] also used students' gaze information to infer the tutors strategy in terms of the instruction and feedback to be given, and the emotions of the tutor. Wang et al. [2006] also used gaze as the interaction modality for students to interact with the system. In a preliminary usability testing Wang et al. [2006] found that such a feedback improved students' involvement with the learning processes. Jaques et al. [2014] used gaze data to predict students' boredom and curiosity for encouraging students to use self-regulated learning strategy.

Gaze awareness was also shown to be effective in improving the quality of online collaboration in two different studies by Oh et al. [2002] and Tan et al. [2009]. The basic idea was to present the collaborating users the gaze information of their partner. Tan et al. [2009] used eye-contact as a proxy for gaze awareness as they placed the camera, capturing users' frontal faces, behind a semi-transparent glass window (which was also their collaborating space) to enable users share eye-contact with their partners without taking their eyes off the display. On the other hand, Oh et al. [2002] conducted a usability study, where they compared three interaction modalities to activate/deactivate a feedback system. The three modalities were: looking at and looking away from the agent to activate and deactivate; pushing a button and giving a voice command. The authors found that the users preferred the gaze interaction modality over the others.

In this chapter, we present an eye-tracking study that gives real-time feedback to the students based on their gaze. The key difference from the previous studies is, that we gave feedback directly to the students rather than providing it to a tutor. The system computes students' with-me-ness levels and gives them a visual feedback on the video lecture, if their with-me-ness levels falls below a certain threshold.

6.3 Problématique

We conducted an eye-tracking study where the participants attended a MOOC lecture and received feedback about what are the places which the teacher is talking about. We used the data collected from the experiment in chapter 6 to create a baseline for students' with-me-ness. Students received feedback whenever their with-me-ness was less than the baseline at any given point of time in the video. The major hypothesis was that the gaze aware feedback will increase students' with-me-ness; and thus their attention during the video lecture. The secondary hypothesis was derived from the first hypothesis. We expected the learning gains to be higher in this experiment than the previous experiment because the students would be paying more attention to the lecture content. Through this study we addressed the following research questions:

1) How does the gaze aware feedback affect the gaze patterns while watching the video?
2) How does the gaze aware feedback affect learning gain of the participants?

6.4 Experiment

6.4.1 Participants and procedure

There were 27 bachelor students from École Polytechnique Fédérale de Lausanne in Switzerland participating in the present study. There were 6 females among the participants. The participants were compensated with an equivalent of CHF 25 for their participation in the study.

Upon their arrival in the laboratory, the participants signed a consent form. Then the participants took a pretest (Appendix D) about the video content. Then the participants watched two videos about "resting membrane potential". Finally, they took a posttest (Appendix F). The videos were taken from "Khan Academy". The total length of the videos was 17 minutes and 5 seconds. One important point worth mentioning here is that the teacher is not physically present in the video. The participants were told that the feedback would appear only when they were not paying attention to what the teacher was saying or writing.

6.4.2 Gaze aware feedback

The feedback was displayed on the screen as red rectangles circumscribing the area of the screen which the teacher was talking about (Figure 6.1). The feedback was shown only when the with-me-ness levels of the participant went below a baseline. This baseline was calculated for each second of the video lecture. To calculate the baseline we took only those participants from the previous experiment whose leaning gain was fell between 33 and 66 percentile of the overall learning gain of the previous experiment. The reason for selecting this range of scores because we wanted to give the feedback based on the typical behaviours of the students from the previous experiment. In the remaining part of this chapter this group is called the "baseline group". To be able to compare the two groups (baseline and experimental) we only considered the "textual" priming group from the experiment mentioned in Chapter 5. The learning gains of the two groups are comparable as they had the same pretest and posttest. We considered only a subset of this group to define our baseline, however, to compare the learning gains we will use the complete set (with 50 students).

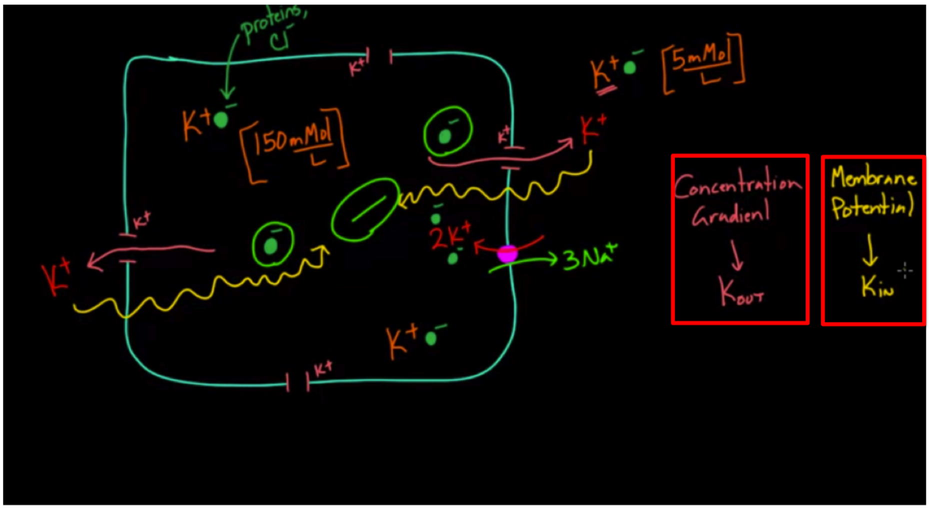


Figure 6.1 – *Example of the feedback used in the experiment. The circumscribing red rectangle were shown if the with-me-ness of the participant went below the baseline with-me-ness at any given instant during the video. For this particular frame, Teacher: "so you have one force, the concentration driving K out; and another force, the membrane potential, that gets created by its absence that?s gonna drive it back in."*

6.4.3 Dependent variables

1) **Learning Gain:** The learning gain was calculated as the difference between the individual pretest and posttest scores. The minimum for each test was 0 and 10, and the maximum for the pretest was 9 and for the posttest was 10.
2) **With-me-ness:** We used the same method as described in Chapter 6, to calculate students' with-me-ness levels, in this experiment, in real time.

6.5 Results

Feedback and Learning Gain: We observed a **significant improvement in learning gain for the experimental group over that for the baseline group** (t (df = 49.88) = -2.50, p = .02, figure 6.2a).

Table 6.1 – *Mean and standard deviations for learning gains across conditions.*

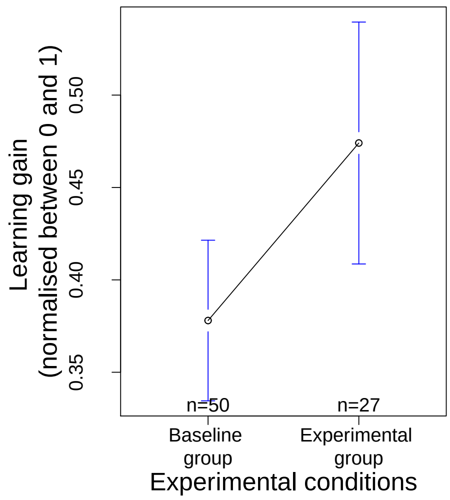
Condition	Number of participants	Mean	Std. dev.
Baseline	50	0.38	0.15
Experimental	27	0.47	0.16

Immediate effect of feedback on Gaze: We observed a significant improvement in with-me-ness levels for participants (within the experimental group) before (mean = 0.31, sd = 0.08) and after (mean = 0.57, sd = 0.16) displaying the feedback (F [1, 26] = 310, p < .001, figure 6.2b). The time difference between the moments before and after displaying the feedback was usually 2 seconds.

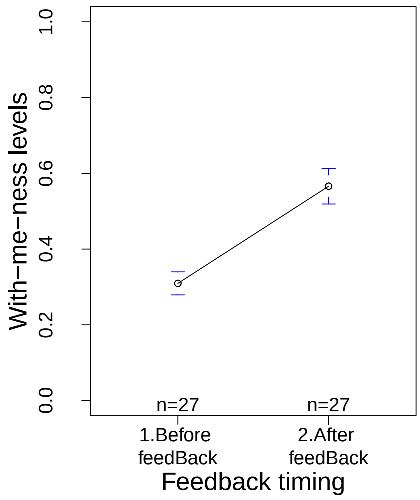
Overall effect of feedback on Gaze: In order to find the overall effect of the feedback on the participants' gaze, we divided the whole video in one minute episodes. Results from a linear mixed effect model showed that on average, participants' with-me-ness increased by 1% every minute. This improvement was significant over time (F [1, 26] = 32.60, p < .0001). Table 6.2 shows the summary of linear mixed effect model with time and participant ID as fixed and random effects respectively. Figure 6.2c shows the temporal evolution for the difference between the mean observed with-me-ness and the baseline with-me-ness for the participants; and the average number of time the feedback was shown to the participants. We can see in figure 6.2c that, towards the end of the video, the difference increased and the number of feedback displays decreased. This showed that the participants became more aware of the fact that they should follow the teacher in an efficient manner in order to learn.

6.6 Discussion

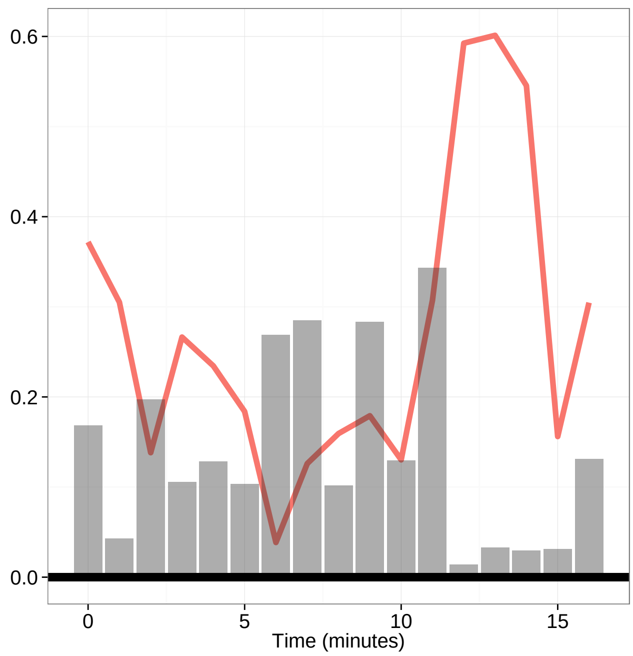
There was a significant improvement in the learning gains for the students in the experimental condition than the baseline condition. We could conclude that the gaze aware feedback helped



(a) *Learning gain for the experimental and baseline conditions.*



(b) *Immediate effect of feedback on with-me-ness.*



(c) *Overall effect of feedback on the gaze. The whole video was divided into one minute episodes. The red curve shows the difference between the observed and baseline with-me-ness (smoothened using a two minute rolling window). The bars denote the number of feedbacks per participant per minute.*

Figure 6.2

Table 6.2 – *Linear mixed effect model with time and participant ID as fixed and random effects respectively.*

	Mean	Std. error	t-value	p-value
Intercept	0.19	0.03	6.94	<.01
Time	0.01	0.002	5.71	<.01

the students to learn more. However, this result has to be treated carefully, as the populations were largely similar (the participant recruitment was done using the same university channel, and there was no drastic changes in student populations) in the two conditions, however the two groups of students were in two different years of the university education (the two studies were conducted one year apart from each other).

We found a significant immediate effect of the feedback on participants' gaze. The with-me-ness levels were significantly higher after showing the feedback than those before showing the feedback. One plausible explanation emerged from the salient nature of the feedback. Since the red rectangles appeared as a salient visual feature for the participants, their attention was drawn towards the feedback.

However, the significant long term effect on the with-me-ness indicates that the feedback had an effect on participants' attention in the terms of "how well they follow the teacher in both the deictic and dialogue spaces". One plausible interpretation of increase in with-me-ness over time, could be, that the participants became more aware of the fact that following the teacher during is important to understand the content and they started following the teacher more closely than before. This effect is also evident from the figure 6.2c. We can see that the difference between the baseline with-me-ness and the observed with-me-ness was higher during the second half of the video.

Concisely, we could say that the gaze aware intervention in the learning process of the students was observed to have a positive effect on their attention. Provided that such a feedback is used during regular MOOC studies, this might have a long term impact on students' overall attention. In terms of our general research question about "how to improve the attention of the students during MOOC videos"; gaze aware feedback emerged as one of the positively influencing intervention.

Our way of providing gaze-aware feedback to students has a key limitation in terms of pre-processing required. The computation of with-me-ness requires us to know all the deictic gestures and to transcribe the dialogues beforehand. This might be overwhelming for longer videos. One way to overcome this issue is to use the heat-maps to convey the content coverage and provide feedback to the students about their gaze patterns.

7 Effect of Displaying the Teacher's Gaze on Video Navigation Patterns

7.1 Introduction

In previous chapters, we have shown the importance of following the teacher in achieving high learning outcomes. The gaze-measure "with-me-ness" was found to be correlated with students' learning outcome. We used the gaze as a measure of attention and a way to provide feedback to the students. The gaze-aware feedback was shown to be effective in terms of both the gaze patterns and the learning gain of students. In this chapter, we addressed a different question; "can we use gaze as a tool to drive attention?" One way to improve students' learning experience could be to make teachers' discourse easy to follow by augmenting additional information on the video lecture. In this experiment, we chose to augment the video lecture with teacher's gaze and use students' navigation patterns to quantify the ease of following teacher's discourse.

To address the question, whether we could use the teachers' gaze to help making the learning process efficient for the students, we augmented the teacher's gaze on a MOOC video on Coursera (this was not an experiment in the lab). We then collected the MOOC logs containing the video navigation patterns; and analysed the data to find the effects of displaying the teacher's gaze on the video navigation patterns of the students.

In this chapter, we show that displaying teacher's gaze in a MOOC video-lecture could help the students understand more easily the content of a MOOC video. Moreover, this effect remains consistent with the increasing complexity of the situation explained by the teacher.

7.2 Context

7.2.1 Gaze contingency and reference disambiguation

We know from previous eye-tracking research that speakers looked at the objects they refer to just before pointing and verbally naming the objects [Griffin and Bock, 2000]. Listeners on the other hand, looked at the referred objects shortly after seeing the speaker point and refer to the objects [Allopenna et al., 1998]. Richardson et al. [2007] showed that the listeners who were better at attending the references made by the speaker were also better at understanding the context of the conversation. One way to aid the listeners attending the reference in a better way could be to display where the speaker is looking at. This might help the listeners in a better disambiguation of the complex references [Gergle and Clark, 2011, Hanna and Brennan, 2007]. In the case of complex stimulus displaying the gaze of speaker made the disambiguation of the references even easier [Prasov and Chai, 2008]. This motivated us to study the effect of showing the gaze of the teacher in a MOOC video on the navigation patterns of the students.

Gaze contingent experiments are at the proactive side of the eye-tracking technology. These experiments consist in displaying the gaze of collaborating partners to each other; or displaying the gaze of an expert to a novice in order to teach the novice [Chetwood et al., 2012] . Another modality of gaze contingency is using gaze as a mode of communication. In a collaborative "Qs-in-Os" search Brennan et al. [2008] showed that the sharing gaze information between collaborating partners resulted in a strategy of division of labour as effective as if the partners were talking face to face. Using gaze as a communication modality Prendinger et al. [2007] used gaze information to inform participants about the effectiveness of grounding process between a human and an infotainment presentation agent. In a multiparty video conference system Vertegaal et al. [2002] used gaze information to rotate the participants' virtual 3D representations to the persons they were talking to. Displaying the gaze of speaker helped the listener in deciphering the references [Gergle and Clark, 2011, Hanna and Brennan, 2007]. Moreover, gaze of speaker made it easier for the listener in deciphering the references in situations with high ambiguity [Prasov and Chai, 2008].

7.2.2 Online video navigation profiles and the perceived difficulty of content

Students' navigation styles could tell us a lot about their perception about the content. In order to find the effect of displaying the teacher's gaze on the students' navigation pattern and in turn their learning experience, we required a proxy variable that could quantify the learning experience. Li et al. [2015] conducted a study with over 30,000 students and 100 videos across two courses where the authors asked students to rate the perceived difficulty of the content after the students watched the video. Based on students' rating and their video navigation behaviour [Li et al., 2015] concluded that the students who perceived the video content as easy to understand did less frequent and shorter pauses, and replayed the video less frequently. We chose to build upon the results from Li et al. [2015], using the students' video navigation patterns, for the video augmented with the teacher's gaze.

7.3 Problématique

We carried out a study in order to explore the effects of displaying gaze of the teacher on the students' video interaction patterns. The teacher's gaze was recorded when he was recording

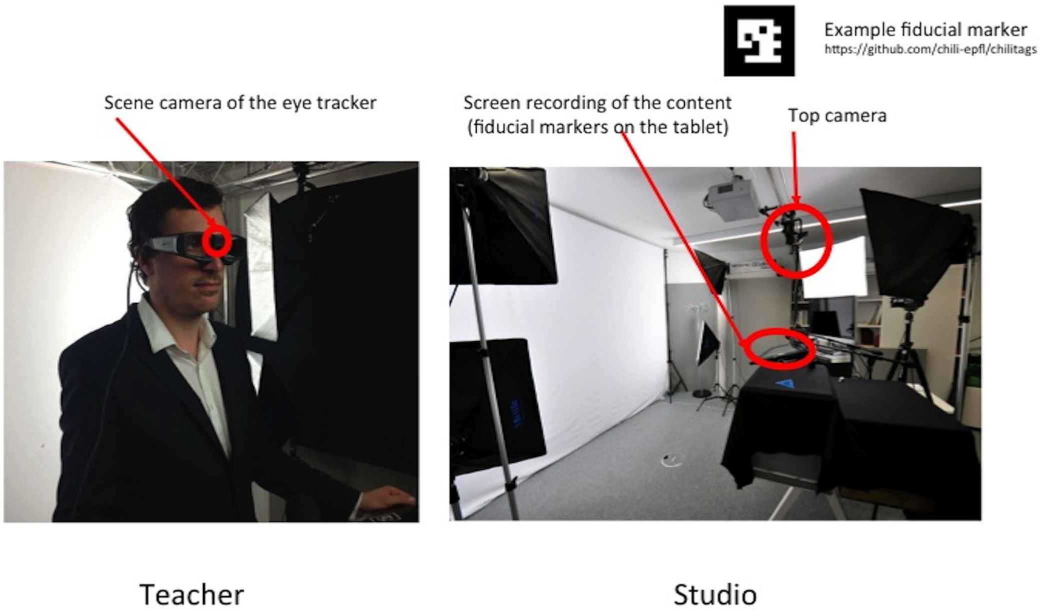


Figure 7.1 – *Setup: The teacher is equipped with the SMI mobile eye-tracking glasses (left) and the MOOC recording studio (right) with the top camera on the ceiling and the tablet used by the teacher. The fiducial markers (top-right) are glued to the tablet to make the re-localisation of teacher's gaze on the actual content easy.*

the MOOC video. Our prime hypothesis was that displaying teachers' gaze on the video would make the reference disambiguation easy in high ambiguous situations. Moreover, displaying teacher's gaze on the video would also make the students' behaviour more linear in terms of following the content (fewer pauses and fewer backward jumps).

7.3.1 Research Questions

Through this experiment, we wanted to explore following two research questions:

1) What is the effect of displaying teachers' gaze on a MOOC lecture on students' video navigation patterns? Our hypothesis is that displaying teacher's gaze on the video would reduce the actions of the students on video display and the students' behaviour would be more linear in terms of following the content, i.e., they would pause and move forward/backward less (*behavioural hypothesis*).

2) If there is a relation between the students' video interaction patterns and teacher's gaze, how is it moderated by the ambiguity of the video? We hypothesise that displaying teachers' gaze on the video would make the reference disambiguation easy in ambiguous situations (*eye-tracking hypothesis*).

7.4 Experiment Setup

We asked one of the teachers to track his eyes on the MOOC video he was going to record. We used SMI mobile eye-tracking glasses to record the gaze of the teacher. The main motivation to use mobile eye-trackers was to give the teacher as ecologically valid environment as possible. The setup of the MOOC recording studio is shown in figure 7.1. The teacher was equipped with the eye-tracking glasses. There was a screen capture software running on the tablet with the actual content to record every move of the teacher. Also, there was a camera on the ceiling of the studio to capture the gestures (external to tablet) on the tablet. We put nine fiducial markers [1] on the tablet so that later we were able to re-locate the gaze pointer of the teacher on the tablet. The video was uploaded on Coursera as one of the video lectures during one of the weeks of the course "Villes africaines: Introduction à la planification urbaine" (African cities : an introduction to urban planning)[2] . The teacher explicitly chose the parts of the video where he wanted to display his gaze.

7.4.1 Re-localisation of teacher's gaze

We recorded three different video streams from the setup of figure 7.1. *1)* the video from scene camera of the eye-tracker. *2)* from the top view camera in the studio. *3)* the video from the screen capture software running on the teacher's tablet. We knew teacher's gaze positions in the frame of the video captured from the scene camera of the eye-tracker. The objective was to find the gaze positions on the video from the screen capture of the tablet. This was not a trivial task. Since the teacher was given full freedom to move, his field of the view of changed at every instant. We computed the gaze positions on the actual content using following steps (figure 7.2):

1) We computed the relative position of the fiducial markers and the gaze positions in the video from the scene camera of the eye-tracker.
2) We computed the relation between the positions of the fiducial markers in the video from the top camera and the video from the scene camera of the eye-tracker.
3) Using the two relations, we computed in steps 1 and 2, we computed the gaze positions on the video from the top camera. The output of this step was a video where the gaze pointers are shown on the video from the top camera.
4) The video from the top camera was geometrically a distorted version of the video from the screen capture software running on the tablet. Hence, we removed the distortion from the resulting video of step 3 to get the video from the screen capture software with teachers' gaze pointers.

[1] Chilitags
[2] The MOOC "Villes africaines: Introduction à la planification urbaine" was given by Prof. Jérôme Chenal. This course was developed at École Polytechnique Fédérale de Lausanne, Switzerland.

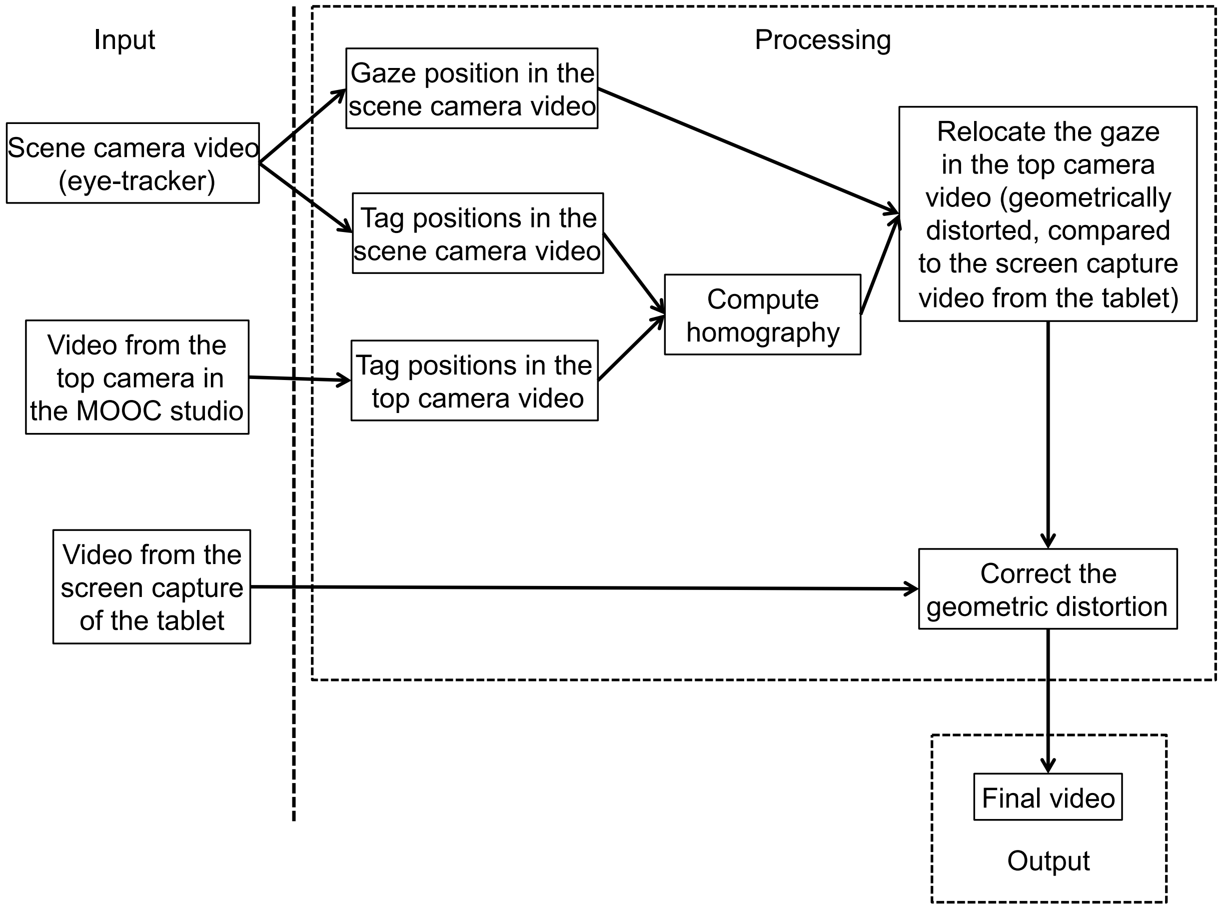


Figure 7.2 – *Process for the re-localisation of the teacher's gaze on the final video output.*

7.4.2 Ambiguity in stimulus and teacher's gaze

To analyse the students' behaviour we divided the video into four episodes based on whether there was teacher's gaze present on the video, and what was the level of ambiguity in the images shown in the video (high vs low ambiguity). The ambiguity in the image was determined by *how easy was it to disambiguate a simple verbal reference on any part of the image.* Simply put, how easy it was to locate what part of image/scene the speaker was talking about. Images with high ambiguity were satellite images and aerial images where the target reference were smaller in size and are not obviously present in front of the listeners' eyes. Whereas, images with low ambiguity were street views where the target references were bigger in size and were easily detectable by the listeners. Examples of images with high and low ambiguity are shown in figures 7.3 and 7.4 respectively. This categorisation was later confirmed by the teacher himself. The main reason for this categorisation was to be able to segment the video in high and low ambiguity stimulus periods.

Figure 7.3 – *Example of a high ambiguity image from the experimental video. The image is an aerial view and the teacher is explaining the landscape captured. We rate these type of images because high ambiguity images as disambiguating a reference like "'the school" is difficult without a visual cue.*

Figure 7.4 – *Example of a low ambiguity image from the experimental video . The image is typical street view and the teacher is explaining the landscape captured. We rate these type of images as low ambiguity images because disambiguating a reference like "'the tree" is easy without a visual cue.*

7.4.3 Measures

In this subsection, we present the measures of students' behaviour we used to analyse the affect of displaying the teacher's gaze in the video. We compared the measures in two ways: *1)* we compared the variables for the experimental video (video with teacher's gaze) and other videos (*between videos variable*); *2)* we compared the values of the variable within the experimental video for different episodes in the video (*within video variable*).

1) **Proportion of replayed video length:** This was calculated by counting the number of video seconds that were played more than once. This supposedly indicated the difficulty that student experiences during the video lecture. A high proportion of replayed video for a student could suggest that the student was not able to understand some of the content properly in the first time going through the video. This was used only as a between video variable.

2) **Frequency of pauses per minute:** This was the average number of pauses that a student makes during one video per minute. High number of pauses might indicate the difficulty or frequent disengagements from the video. This variable was used as both a between and within video variable.

3) **Ratio of pause time and video length:** This was the total time spent by the students while keeping the video in a pause state divided by the total video length. Longer pauses would result in a higher value of the ratio. Moreover, the higher ratio might indicate the difficulty in understanding the video as students would need more time to grasp the concept. This variable was used only as a between video variable.

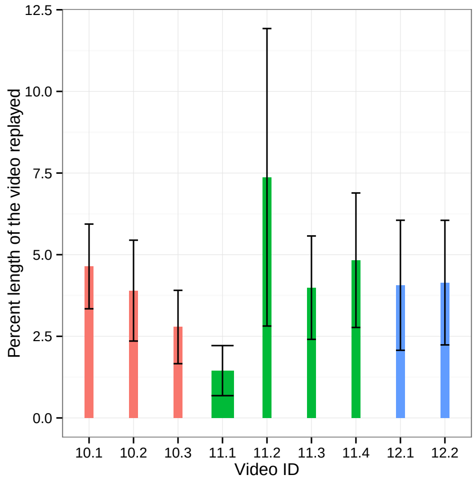
4) **Frequency of seek backs per minute:** This was the average number of backward jumps that a student makes during one video. The seek back event typically reflected two necessities from a student. First, a check for a reference that was made at a previous video point. Second, a complete section of the video being too difficult to understand and the student decided to re-watch the whole video segment. This variable was used as both the between and within video variable.

7.5 Results

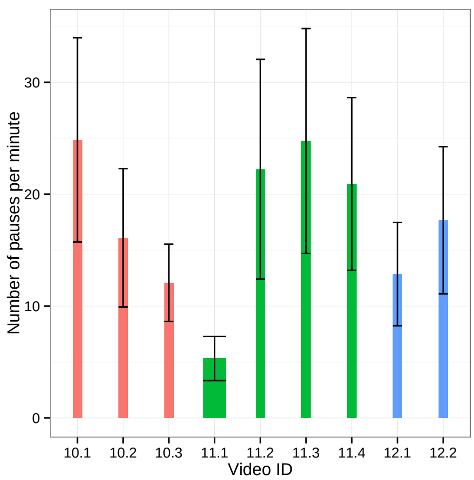
As we mentioned in the section 7.4.3, there were two levels of analysis to be presented: *1)* we compared students' behaviour across different videos in the weeks succeeding and the preceding the week of the experimental video; *2)* we compared the students' behaviour across different episodes within the experimental video. The three weeks were weeks 10, 11 and 12, which also were the last weeks of the course. The main reasons behind selecting only three weeks to compare were that, the size of student population was comparable for these three weeks; and that the population was comparable in terms of the motivation to finish the course and the levels of engagement.

7.5.1 Comparing user behaviour across different weeks

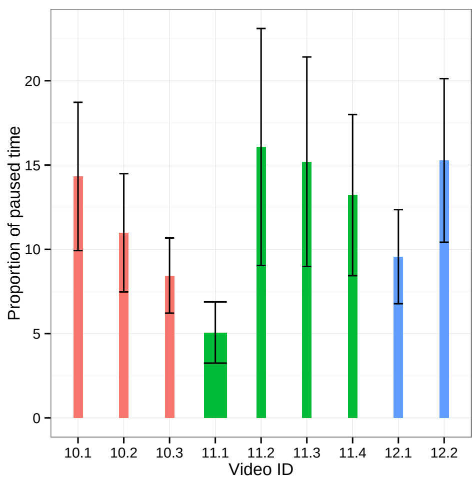
In this subsection, we compared the number of pauses, seek backs, seek forwards, the pause time and replay time across different videos. The experimental video is labeled as "11.1". In the figures 7.5a - 7.5d the variables corresponding to the experimental video are shown as a thicker bar than the other videos.



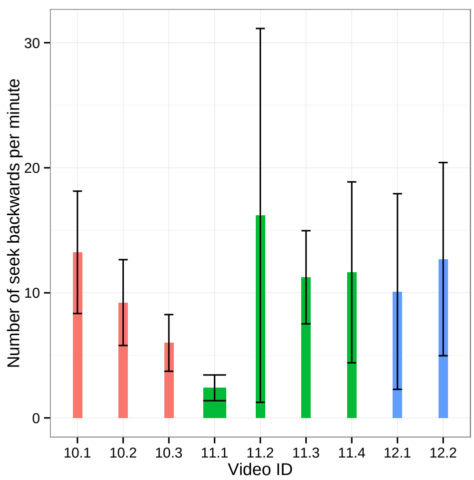
(a) *Proportion of replayed video length.*



(b) *Average number of pauses.*



(c) *Ratio of pause time and video length.*



(d) *Average number of seek back events.*

Figure 7.5 – (a) Proportion of replayed video length, (b) Average number of pauses, (c) Ratio of pause time and video length, and (d) Average number of seek back events; compared across weeks 10, 11 and 12.

1) **Proportion of replayed video length:** An ANOVA with the lecture ID as a between subject factor showed that the proportion of the replayed length video was the lowest (figure 7.5a) for the experimental video ($F[9,4202] = 2.12$, $p = .03$).

2) **Frequency of pauses per minute:** An ANOVA with the lecture ID as a between subject factor showed that the average number of pauses was the lowest (figure 7.5b) for the experimental video (F[9,4202] =2.89, p = .002).

3) **Frequency of seek backs per minute:** An ANOVA with the lecture ID as a between subject factor showed that the average number of seek backs was the lowest (figure 7.5d) for the experimental video (F[9,4202] =1.92, p = .04).

4) **Ratio of pause time and video length:** An ANOVA with the lecture ID as a between subject factor showed that the ratio of pause time and video length was the lowest (figure 7.5c) for the experimental video (F[9,4202] =2.58, p = .005).

7.5.2 Comparing user behaviour within the video

In this subsection, we compared the number of pauses, and seek back actions for different episodes within the experimental video (figures 7.6 and 7.7). As we explained in section 7.4.2, the experimental video was divided in 4 different kinds of episodes based on two facts: *1)* whether teacher's gaze is present or not; and *2)* whether the ambiguity in the video was high or low. One might argue that the teacher deliberately chose the moments, to display his gaze, where the ambiguity was highest. However, we did not find any significant difference between the lengths of the four different episodes ($\chi^2(df = 1) = 0, p > 0.5$, table 7.1).

Table 7.1 – *Lengths (in minutes, chi-square residuals in parentheses) of the different episodes within the experimental video. Residuals (absolute values) more than 1.96 are considered to be significant.*

	High ambiguity	**Low ambiguity**
Gaze-present	2.46 (0.25)	2.46 (-0.21)
Gaze-absent	5.20 (-0.15)	7.80 (0.13)

In the table 7.2, we observed the following:

Number of pauses in "gaze-present" episodes was lower than that in "gaze-absent" episodes. Moreover, there were lower number of pauses in the high ambiguity situations than those in low ambiguity situations (χ^2 = 79.83, p < .001).

Number of seek backs in "gaze-present" episodes was lower than that in "gaze-absent" episodes. Moreover, there we're lower number of seek backs in the high ambiguity situations than those in low ambiguity situations (χ^2 = 164.83, p = .001).

7.6 Discussion

The results in section 7.5.1 showed that the behavioural hypothesis (section 7.3.1) stands true. The fact that the students had fewer seek back events could reflect the fact that they did not need to check back the previously told content because it was easy to understand for them

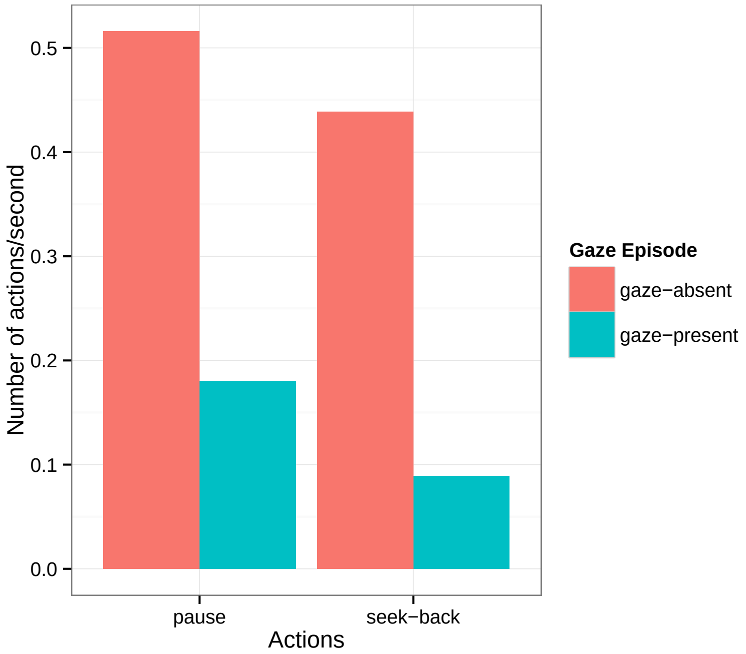


Figure 7.6 – *Proportions of different types of events compared within the experiment video across different gaze episodes.*

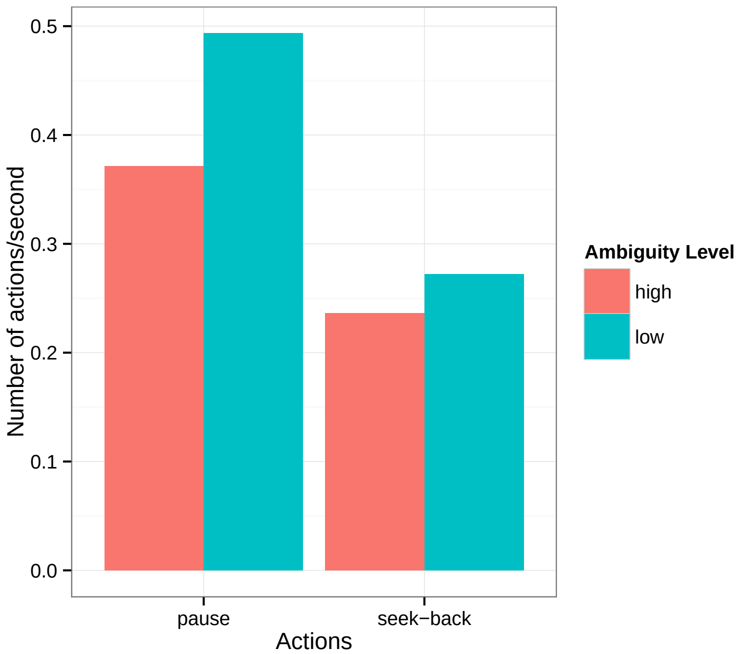


Figure 7.7 – *Proportions of different types of events compared within the experiment video across different ambiguity episodes.*

Table 7.2 – *Numbers (chi-square residuals in parentheses) of different types of events, for the different episodes within the experimental video. Residuals (absolute values) more than 1.96 are considered to be significant.*

	Actions			
	Pause		**Seek-back**	
	High ambiguity	**Low ambiguity**	**High ambiguity**	**Low ambiguity**
Gaze-present	16 (7.22)	64 (-4.27)	18 (-4.27)	23 (-5.71)
Gaze-absent	94 (-2.97)	232 (-2.97)	52 (1.21)	142 (8.77)

once the teacher's gaze was displayed on the video. Moreover, the same fact was strongly supported with less amount of video content replayed for the experimental video. Similarly, less frequent and shorter pauses could indicated that the content delivery was also easy due to the presence of additional cues to disambiguate complex references during the video. Li et al. [2015] found similar video navigation patterns in their study for the students who perceived the video content as easy to understand.

The observation that there were fewer seek back and pauses during the experimental video also verifies our working hypothesis that gaze contingency made the learning experience more linear as compared to the video material. With less breaks in the content delivery and the less back references the students were well aligned with the video content in the temporal space and hence the understanding the content for the student could be effective and efficient.

The key difference between the experimental video and the videos from the other week was the augmentation of teacher's gaze on top of the video content. Since the students could see where the teacher was looking and it had been proved by eye-tracking research that people started looking at the point they were about to refer and hence it was easy to disambiguate the point of reference for the listener when (s)he saw the gaze of referee.

The results from the section 7.5.2 proved our eye-tracking hypothesis (section 7.3.1) to be true as well. The students had fewer pause and seek backs in high ambiguity situations, such as the teacher describing complex images like satellite captured image (figure 7.3),when the gaze was present in the video as compared to when the gaze was absent on the video. This effect is present, although less pronounced, in situations with low ambiguity (for example when the teacher was explaining a street view, figure 7.4). Prasov and Chai [2008] also found in their study about reference disambiguation in complex stimulus that the displaying the gaze of speaker made it easy for the listener to disambiguate the reference.

Although the results supported our hypothesis, more experimentation is required to find out whether displaying teacher's gaze helps in increasing the effectiveness of learning experiences. Moreover, further investigation is necessary to comment on the effect of augmenting multiple MOOC videos with teacher's gaze on the overall learning experience of students.

The introduction of teachers' gaze might also work as a novelty in the engagement process of the students as well. To keep the engagement up to a level which benefits the student, such novelties could prove to be effective. The results showed that usually during the end of the course the students who watch the videos decreased drastically. However, once we put the experimental video online the number of students who watched the video increased from the previous week.

In a nutshell, both of our hypotheses were verified, and this could be an interesting continuation to experiment with augmenting the MOOC videos with the visual cues to help students better understand the content. Our future work includes experimentation with different eye-tracking data visualisations to augment the MOOC video and check how it affects the students' video navigation patterns and their learning processes. Also to perform a laboratory experiment to see how closely the students follow the gaze pointer of the teacher and how it affects their learning outcome.

8 General Discussions

8.1 Scaling up the results

As we said in the introduction, the eye-tracking results could be scaled up from the laboratory experiments to the population scale of MOOCs. One way to scale up the results was to find a common variable, in both the lab experiments and in the MOOCs, such as video navigation patterns. In the same experiment, as in Chapter 5, we found that both the levels of with-me-ness were negatively correlated (for perceptual with-me-ness Pearson correlation = -0.30, $p < .001$ and for conceptual with-me-ness Pearson correlation = -0.53, $p < .001$) to the amount of time spent on a given episode of the Massive Open Online Courses (MOOC) video. Figure 8.1 shows the temporal evolution of the perceptual and conceptual levels of with-me-ness and the time spent on each 10 second episode of the video. We can see that when the students spent more time on a particular video segment, their average with-me-ness was lower. One plausible explanation could be the fact that when students did not pay much attention to the teacher, i.e., when they have low with-me-ness, they had to go back to the video segment at least once more in order to revise the content. Thus, the average with-me-ness was lower.

Also we found out that there was a relation to video playback time and students' performance in MOOCs. Although, making a direct and strong claim about the relationship of hypothetical with-me-ness during MOOCs and students' performance could not be possible; however, video navigation patterns could provide a fair proxy for the gaze data. Moreover, conducting such experiments for a bigger population could also be possible in near future with the cost of high quality eye-tracking systems dropping rapidly.

8.2 Roadmap of results

We conducted several studies to understand the underlying processes of ongoing collaboration and MOOC learning. Following are the summary of main results:

1) **Pair Program Comprehension.** We found that the gaze, dialogues and comprehension

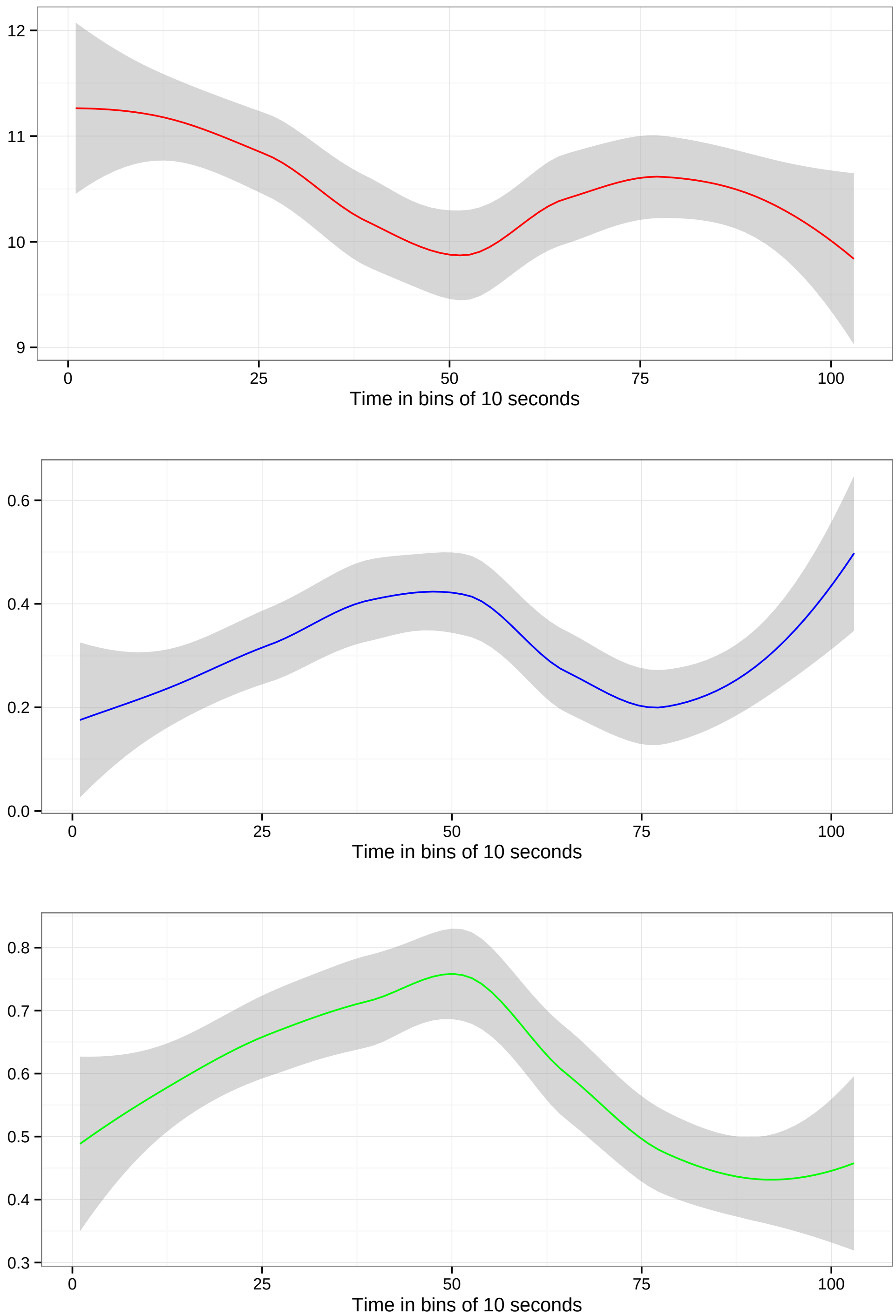


Figure 8.1 – *Temporal evolution of perceptual (green curve) and conceptual (blue curve) levels of with-me-ness and the time spent (red curve) on each 10 second episode of the video. The grey area shows the confidence intervals for 98 students.*

share a triumvirate relation. In terms of gaze and understanding, we found a high correlation between the pairs' gaze similarity and their attained level of understanding. In terms of dialogues and understanding, we found that the pairs having higher level of abstraction in their description of the program functionality attain a higher level of understanding. Also, when the abstraction in the description was higher, the gaze was often directed towards the main functions of the program; rather than guessing the program functionality from the interface messages.

2) **Exploratory MOOC study.** We proposed a new gaze measure to compute students' engagement with the teacher during a video lecture called with-me-ness. We found that with-me-ness, both at the deictic and dialogue levels, was correlated to students' learning outcome.
3) **Dual eye-tracking study with MOOCs.** We found that the individual gaze patterns during the video lecture were correlated to the collaborative gaze patterns during a collaborative concept map activity. Moreover, we also found that it was possible to shape students' attention to a particular part of the video by using different representations of the same content as priming.
4) **Gaze-aware feedback.** We designed a gaze-aware feedback tool based on students' with-me-ness to support them pay attention to the correct areas during a MOOC video. We found that the feedback had positive effects on the with-me-ness and the learning gains of students.
5) **Displaying teacher's gaze on MOOC video.** We found that displaying teacher's gaze on the MOOC video helped students disambiguate the references easily and hence the students perceived the video easier than when there was no gaze displayed on the video.

8.3 Contributions

In this section, we discuss the contributions of this dissertation within the relevant research areas.

8.3.1 Eye-tracking and learning analytics

Eye-tracking had been shown useful (as described in "Related Work") for differentiating performance levels, task difficulty, and expertise. We defined new eye-tracking variables in order to capture these differences in more details. The variable with-me-ness not only captured the moments of explicit referencing but also the verbal/implicit referencing. We considered the student watching the MOOC videos as interacting with the teacher, to ground our findings using with-me-ness and then showed that it could be used to design an effective gaze-aware feedback tool for MOOC learners. Thus completing the learning analytics loop (figure 8.2) as a cybernetic control system.

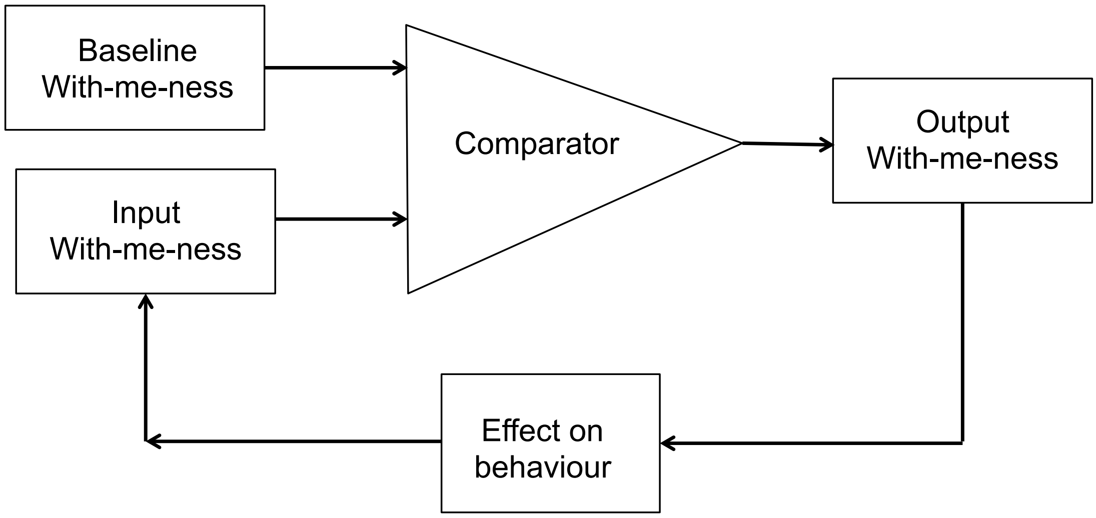


Figure 8.2 – *The cybernetic control (learning analytics) loop using with-me-ness.*

8.3.2 Interaction styles

We also showed through different experiments that there were basic differences in how people interacted with a visual stimulus. Those having high with-me-ness and high gaze similarity look “through” the stimulus and interacted with the teacher/collaborating partner. On the other hand, those who had low with-me-ness and low gaze similarity, looked “at” the stimulus and interacted with the content only. This was also true for program comprehension. Pairs who followed the data flow of the program look “through” the stimulus and understand the logic of the program in an efficient manner. Whereas, those who read the program as a piece of text looked “at” the program and had difficulties understanding it. This notion of looking “at’ or looking “through” goes beyond learning context and could also be exemplified within the context of psychiatry. The *art brut or outsider art* is one such example, where the art pieces created by psychologically challenged persons could provide supports to psychiatrists, which are beyond the pathological symptoms. This is an example of looking “through” the art work and interacting with the artist.

8.3.3 Collaborative problem solving

The dual eye-tracking research had been highly task dependent. The main problem we addressed in this thesis was how to automatically segment the interaction irrespective of the task at hand. We showed that the gaze similarity episodes could be computed for any kind of task, because it does not depend on the basic properties of the problem and the stimulus. In this thesis, we used them for program comprehension and concept map tasks. There are inherent differences in the visual nature of the two tasks. In the case of program comprehension the content is static and it has the same visual structure for the participants. However, the concept maps are dynamic in nature and the visual structure can be different for different teams.

We also showed that the abstraction in the dialogues was closely related to the gaze patterns in a collaborative problem solving situation. These results could be leveraged upon to use the gaze data as a proxy for the dialogues in a collaborative setting as the success in automatic analysis of dialogues is bounded by the current limitations of Natural Language Processing (NLP) algorithms.

8.4 Design implications from the studies

In this section, we present the design guidelines, to analyse, and/or develop an intelligent agent to support dyadic interaction. These guidelines are based on the relationships between the gaze patterns, the dialogues, and the level of success attained by the dyad/individual after the interaction.

Considering the guidelines for designing an intelligent agent to support program comprehension. This could be important for those working with "legacy softwares", where the new programmer might not have been a part of the original development team. In our pair program comprehension experiment, we observed a strong relation between people following the data-flow of the program and their comprehension levels. In individual settings, one can use the data-flow to make the comprehension process easier. These data-flow patterns could be highlighted by the program editor itself. Moreover, in collaborative settings, one could observe the abstraction in the dialogues, in addition to highlighting the data-flow. In our study, we observed a strong relationship between the abstraction in the dialogue and the level of comprehension attained by the pair. In the terms of natural language processing, the abstraction in the dialogues is easier to capture, than other features. The abstraction in the dialogues can be captures by simply looking at the proportion of the utterances in the domain language.

Regarding MOOC platforms, we showed in our experiments that, we could capture attention in a seamless (independent, on the student side, of the video content) way by using with-me-ness. We also have shown one application, how this variable could be used to improve both the learning gains and the attention levels of the students. There could be another possibilities for providing the gaze-aware feedback to the students. The heat-maps and scan-paths (which are essentially independent of the semantics of the content) could be used to give the feedback about the content coverage and the missed content to the students. These two variables are easily computes and they do not require high quality and high precision eye-trackers to collect the data as well. Moreover, using the scan-path to compete the missed parts of the lecture, one could provide feedback to the students simply by highlighting them.

Moreover, as we have shown in both the pair program comprehension and collaborative concept map tasks, the gaze similarity could be computed in a semantically independent (from the content) way in real time. One could utilise this variable to improve the collaboration outcome of poor performing collaborators by providing feedback or by simply telling the collaborating partners, where the other partner is looking at.

Throughout the studies presented in this thesis, we have shown that the being together, with the teacher and/or with the collaborating partner, resulted in a better shared understanding or a higher learning gains. But this is not always the case, for example in collaborative visual search, togetherness can be detrimental for task based performance. There is a implicit need to divide the visual stimulus into different parts by the partners [Brennan et al., 2008]. Measuring togetherness could also improve the performance in such situations, where we can give feedback to the collaborating partners about their togetherness, but in a reversed manner. We could alert the pair when their gaze togetherness is higher than a given threshold. In Chapter 6, We supported the students by giving them the feedback on the lack of togetherness. In cases where togetherness is "harmful" we can provide feedback on the excess of togetherness.

One might argue that a few of the results we reported are obvious, for example "if one pays attention to the teacher, (s)he learns better". However, we also showed that, measuring a variable like "how much one pays attention to the teacher" is not a trivial task and also providing the students with such a feedback improves their learning gains. These studies did not only gave us a measure the attention of the students in an automatic manner, but also enabled us to design systems to support students while they follow the MOOC lecture. Moreover, these findings can be extended to the vast population of MOOC students, as we showed in the Section 8.1, using other variable as a proxy to the gaze patterns.

8.5 Limitations and future work

In this section, we discuss some of the limitations with our methods we used in our research. The research was done on a very small sample size in terms of videos, however we made sure that different types of videos are included. The number of MOOCs we experimented could also put limits to the generalisability of results from gaze-contingent experiment.

Another point that could be argued upon is "what is the best method for augmenting deixis on MOOC videos?" The gaze is an efficient way to convey the teachers' references; but the question that, "is it better than having the teacher simply point at the referred cite", still remains unanswered. This could be a possible extension to this dissertation work. Also, how does gaze-contingent videos affect students' long term engagement in a MOOC could also be an interesting direction of work.

The two interaction styles (looking through and looking at) we proposed, needs more formalisation. One can investigate the personality, attitude and learning strategy factors affecting the choice between looking "through" and looking "at". Moreover, the gaze variable "with-me-ness" does not capture raw speech features which might affect the gaze of the listener. This might be another addition to the definition of "with-me-ness".

From the point of view of scaling up the findings of this thesis, to a vast MOOC population requires cheaper and more intelligent eye-trackers (for example, a webcam based eye-tracker). This could be another branch of investigation stemming out from this dissertation. Moreover, from a usability point of view, experimentation is required to study the acceptance of webcam based eye-tracking or cheap eye-trackers embedded in the laptops.

8.6 Final words

This dissertation presented the outcome of a few years of research during which, *1)* a dual eye-tracking for pair program comprehension was conducted. Based upon the findings of which, *2)* we focused our investigation to a special dyad, the teacher-student pair. This simplified the leader-follower question for us. Finally, *3)* as two applications of our findings, we showed that both gaze-contingency and gaze-awareness could have positive effects on learning processes and learning outcome. This thesis could inspire a few different directions to focus the investigation upon. Both the gaze-contingency and gaze-awareness can be further investigated for long term effects on learning. Also, there is also room for other additions in the gaze variables we propose.

A Program used in the pair program comprehension task

```
package tictactoe;

import java.io.BufferedInputStream;
import java.util.LinkedList;
import java.util.List;
import java.util.Scanner;

public class AddThemUp {

       public static interface AddThemUpUi {

      // input
      public int getPlayerMove(int p);
      public int getNewPlayerMove(int p);

      // output
      public void currentGameState(List<Integer> leftNumbers, List<Integer>[]
        playersNumbers);
      public void gameStarted();
      public void badMove();
      public void playerWin(int p);
      public void gameDraw();
      public void gameEnded();

        }

        // cell content is the player number or 0 if empty
        LinkedList<Integer> leftNumbers = new LinkedList<Integer>();
        LinkedList<Integer>[] playersNumbers = new LinkedList[2];

        AddThemUpUi ui;
```

```
public AddThemUp(AddThemUpUi ui) {
        super();
        this.ui = ui;
        playersNumbers[0] = new LinkedList<Integer>();
        playersNumbers[1] = new LinkedList<Integer>();
}

public void run() {
        initGame();
        int currentPlayer = 1;
        ui.gameStarted();
        ui.currentGameState(leftNumbers, playersNumbers);
        int winner = 0;
        while ((winner = checkForWinner()) == 0 && !gameFinished()) {
                int pos = ui.getPlayerMove(currentPlayer);
                while (!checkAndSet(pos, currentPlayer)) {
                        ui.badMove();
                        pos = ui.getNewPlayerMove(currentPlayer);
                }
                ui.currentGameState(leftNumbers, playersNumbers);
                currentPlayer = 2 - currentPlayer + 1;
        }
        if (winner != 0)
                ui.playerWin(winner);
        else
                ui.gameDraw();
        ui.gameEnded();
}

private void initGame() {
        for (int i = 1; i < 10; i++) {
                leftNumbers.add(i);
        }
}

private boolean checkAndSet(int val, int player) {
        if (!leftNumbers.contains(val))
                return false;
        playersNumbers[player - 1].add(val);
        leftNumbers.removeFirstOccurrence(val);
        return true;
}

private int checkForWinner() {
        for (int p = 0; p < 2; p++) {
                for (int i1 = 0; i1 < playersNumbers[p].size(); i1++) {
```

```
                    for (int i2 = i1 + 1; i2 <
                      playersNumbers[p].size(); i2++) {
                            for (int i3 = i2 + 1; i3 <
                              playersNumbers[p].size(); i3++) {
                                    if (playersNumbers[p].get(i1) +
                                      playersNumbers[p].get(i2) +
                                      playersNumbers[p].get(i3) == 15)
                                            return p + 1;
                            }
                    }
            }
        }
        return 0;
}

private boolean gameFinished() {
        return leftNumbers.size() == 0;
}

public static void main(String[] args) {
        AddThemUpUi ui = new AddThemUpUi() {
                private Scanner scanner = new Scanner(new
                  BufferedInputStream(System.in), "UTF-8");
                @Override
                public void badMove() {
                        System.out.println("Bad or already taken
                          number!");
                }

                @Override
                public void currentGameState(List<Integer> leftNumbers,
                  List<Integer>[] playersNumbers) {
                        System.out.println();
                        System.out.print("Numbers left:");
                        for (Integer v : leftNumbers) {
                                System.out.print(" " + v);
                        }
                        System.out.println();
                        for (int p = 0; p < 2; p++) {
                                System.out.print("Player " + (p + 1) + "
                                  numbers:");
                                for (Integer v : playersNumbers[p]) {
                                        System.out.print(" " + v);
                                }
                                System.out.println();
                        }
                        System.out.println();
```

```
                }

                @Override
                public void gameDraw() {
                        System.out.println("Draw! No one wins...");
                }

                @Override
                public void gameEnded() {
                        System.out.println("Game is finished...");
                }

                @Override
                public void gameStarted() {
                        System.out.println("Game has started...");
                }

                private int readMove() {
                        return scanner.nextInt();
                }

                @Override
                public int getNewPlayerMove(int p) {
                        System.out.println("Player " + p + " chooses a
                          number again:");
                        return readMove();
                }

                @Override
                public int getPlayerMove(int p) {
                        System.out.println("Player " + p + " chooses a
                          number:");
                        return readMove();
                }

                @Override
                public void playerWin(int p) {
                        System.out.println("Player " + p + " wins!");

                }
        };
        new AddThemUp(ui).run();
    }
}
```

B Pretest used for the exploratory eye-tracking study for MOOCs

Question 1:

```
public class Test {
   public static void function (int N) {
      if (N == 1) {
         System.out.print ( N );
      }
      else {
         System.out.print ( N + " " );
         function ( N - 1 );
         System.out.print( " " + N );
      }
   }
}
```

What will be the output of **function (6)**?

1) 6 5 4 3 2 1 1 2 3 4 5 6
2) 6 5 4 3 2 1 2 3 4 5 6
3) 6 5 4 3 2 1 0 1 2 3 4 5 6
4) 6 5 4 3 2 1 0

Question 2:

```
public class Test {
   public static int function (int N , int K) {
      if (N == 0 || N == K) {
         return ( 1 );
      }
      else {
```

```
            return (function (N - 1 , K) + function (N - 1 , K - 1));
        }
    }
}
```

What will be the output of **function (4, 2)**?

1) 2
2) 4
3) 6
4) 8

Question 3:

```
public class Test {
    public static int function (int N) {
        if (N % 10 == N) {
            return ( N );
        }
        else {
            return (function (N / 10) + (N % 10));
        }
    }
}
```

What does "function" do?

1) adds the last digit to the number
2) adds the first digit to the number
3) adds the first and the last digits of the number
4) adds the digits of a number

Question 4:

```
public class Test {
    public static int function (int a , int b) {
        if (a >= b) {
            return ( 0 );
        }
        else {
            return (function (a + 1 , b) + a);
        }
    }
}
```

What does “function” do?

1) adds the two numbers a and b
2) adds the numbers from a to b (excluding a)
3) adds the numbers from a to b (both including)
4) adds the numbers from a to b (both excluding)

Question 5:

```
public class Test {
   public static int function (int a) {
      if (a == 0) {
         return ( 1 );
      }
      else {
         return ( function (a + 1) );
      }
   }
}
```

What is wrong in “function”?

1) infinite recursion
2) no base case for recursion
3) syntax
4) nothing is wrong

Question 6:

```
public class Test {
   public static int function (int a) {
         return ( function (a - 1) );
   }
}
```

What is wrong in “function”?

1) infinite recursion
2) no base case for recursion
3) syntax
4) nothing is wrong

Question 7: Which of the following functions checks **recursively** whether a given number is even?

1)

```
public class Test {
   public static boolean function (int a) {
      if (a == 0) {
         return ( true );
      }
      else if (a == 1){
         return ( false );
         }
         else {
            return ( function (a - 2) );
         }
   }
}
```

2)

```
public class Test {
   public static boolean function (int a) {
      if (a == 0) {
         return ( false );
      }
      else if (a == 1){
            return ( true );
         }
         else {
            return ( function (a - 1) );
         }
   }
}
```

3)

```
public class Test {
   public static boolean function (int a) {
      if (a == 0) {
         return ( false );
      }
      else if (a == 1){
            return ( true );
         }
         else {
            return ( function (a - 5) );
         }
   }
}
```

4)

```
 public class Test {
   public static boolean function (int a) {
```

```
      if (a % 2 == 0) {
         return ( true );
      }
      else {
         return ( false );
      }
   }
}
```

Question 8: Which of the following functions **recursively** counts digits in a number ?

1)

```
public class Test {
   public static int function (int a) {
      if (a == 0) {
         return ( 1 );
      }
      else {
         return ( 1 + function (a / 10) );
      }
   }
}
```

2)

```
public class Test {
   public static int function (int a) {
      if (a % 10 == a) {
         return ( 2 );
      }
      else {
         return ( 2 + function (a / 100) );
      }
   }
}
```

3)

```
public class Test {
   public static int function (int a) {
      return ( (int) Math.ceil(Math.log10(n)) );
   }
}
```

4) None of the above.

Question 9:

```
public class Test {
   public static int function (String s) {
      if (s.length() <= 2) {
         return ( 0 );
         }
      else if (s.substring(0,2).equals("11")){
         return ( 1 + function (s.substring ( 2 )) );
         }
         else {
            return ( function (s.substring ( 1 )) );
            }
      }
}
```

What will be the output of **function("11231114")**?

1) 1
2) 2
3) 3
4) 4

Question 10:

```
public class Test {
   public static int function (int a) {
      if (a <= 1) {
         return ( a );
      }
      else {
         return ( a + function (a - 1) );
      }
   }
}
```

What will be the output of **function(120)**?

1) 120
2) 256
3) 32
4) 16

C Posttest used for the exploratory eye-tracking study for MOOCs

Question 1: Which of the following functions is tail recursive?

1)

```
public int f (int n) {
  if (n == 0) {
     return (0);
  }
  else {
     return (1 + f(n / 10));
  }
    }
```

2)

```
public int f (int a, int b) {
     if (b == 0) {
        return (a);
     }
     else {
        return f(b, (a % b));
     }
  }
```

3)

```
public int f (int a, int b) {
     if (a <= b+1) {
        return (0);
     }
     else {
        return (a + f(a, ((a + b) / 2)));
     }
  }
```

4) None of the above

Question 2: Which of the following functions is tail recursive?
NOTE: in Scala def f (a : Int, b : Int) : Int is same as in Java int f (int a, int b).

1)
```
def f (a : Int, b : Int) : Int = {
     if (b < = 1) {
        return (a);
     }
     else {
        return f((a * b), (b - 1));
     }
  }
```

2)
```
  def f (a : Int) : Int = {
     if (a == 0) {
        return (0);
     }
     else {
        return ((a % 10) + f(a / 10));
     }
  }
```

3)
```
def f (a : Int) : Int = {
     if (a == 0) {
        return (0);
     }
     else if ((a % 10) == 7) {
        return (1 + f(a / 10));
        }
        else {
           return (f(a / 10));
        }
  }
```

4) None of the above.

Question 3:

```
public class Test{
  public int f (int n) {
     return loop(n, 1, 0, 1);
  }
  public int loop (int a, int b, int c, int d) {
      if (a == b) {
        return (d);
      }
      else {
        return (loop(a, b + 1 , d , c + d) );
```

```
      }
   }
}
```

what would be the output of **f(5)**?

1) 2
2) 5
3) 8
4) 10

Question 4:

```
 def function(n : Int) : Int = {
   def loop ( acc : Int, n : Int) : Int = {
      if (n==0) acc
      else if (n%100 == 88) loop (acc+3 , n/100)
          else if (n%10 == 8) loop (acc+1 , n/10)
              else loop (acc , n/10)
   }
   loop (0,n);
}
```

what would be the output of **function(808818)**? NOTE: in Scala def f (n : Int) : Int is same as in Java int f (int n)

1) 2
2) 3
3) 4
4) 5

Question 5:

```
 public class Test {
   public int function(int n) {
      return loop(0,n);
   }
   public int loop(int acc, int n) {
      if (n==0) {
         return acc ;
      }
      else if (n%10 == 7) {
            return loop (acc+1 , n/10);
          }
          else {
            return loop (acc, n/10);
```

```
        }
    }
}
```

What does "function" do?

1) checks if the last digit in a number is 7
2) checks if the first digit in a number is 7
3) checks if the second last digit in a number is 7
4) counts the number of digits in a number that are 7

Question 6:

```
  public double f(double x, double y) {
     return loop (0, x, y);
  }
  public int loop (double acc, double x, double y) {
     if (y < x+1) {
        return acc;
     }
     else {
        return (loop (acc + 1, x, (x + y) / 2));
     }
}
```

what would be the output of **f(2,5)**?

1) 1
2) 2
3) 3
4) 4

Question 7: Which of the following functions is tail recursive?

1)

```
public int f (int n) {
    if (n <= 1){
       return (n);
    }
    else {
       return (f(n - 1) + f(n - 2));
    }
 }
```

2)

```
public int f (int a, int b) {
   if (n <= 1){
```

```
        return (n);
     }
     else {
        return (f(n - 1) * f(n - 2));
     }

  }
```

3)

```
  public int f (int a, int b) {
     if (a == b || b == 0){
        return (1);
     }
     else {
        return (f(a - 1, b) + f(a - 1, b - 1));
     }
  }
```

4) None of the above

Question 8: Here is a recursive function:

```
int triangle(int rows) {
  return (rows == 1)? rows : (rows + triangle(rows - 1));
}
```

Which of the following is the correct tail recursive version of above function?

1)

```
 public class Test {
  public int triangle(int rows) {
     return loop(1,rows);
  }
  public int loop (int acc, int rows) {
     return (rows == 1)? acc : (loop (acc + rows, rows - 1));
  }
}
```

2)

```
 public class Test {
  public int triangle(int rows) {
     return loop(0,rows);
  }
  public int loop (int acc, int rows) {
     return (rows == 1)? acc : (loop (acc + rows + 1, rows - 1));
  }
}
```

3)

```
public class Test {
  public int triangle(int rows) {
    return loop(0,rows);
  }
  public int loop (int acc, int rows) {
    return (rows == 1)? acc : (loop (acc + rows - 1, rows - 1));
  }
}
```

4) None of the above

Question 9: Here is a recursive function:

```
def f(n : Int) : Int = {
  if (n==0) 1
  else Math.pow(n,2)+f(n-1)
}
```

Which of the following is the correct tail recursive version of above function?

1)

```
def f(n : Int) : Int = {
  def loop (acc : Int, n : Int) : Int = {
    if (n==0) acc
    else loop (acc*2,n-1)
  }
  loop (0,n)
}
```

2)

```
def f(n : Int) : Int = {
  def loop (acc : Int, n : Int) : Int = {
    if (n==0) acc
    else loop (Math.pow(acc,2), n-1)
  }
  loop (0,n)
}
```

3)

```
def f(n : Int) : Int = {
  def loop (acc : Int, n : Int) : Int = {
    if (n==0) acc
    else loop (acc+Math.pow(n,2), n-1)
  }
  loop (2,n)
}
```

4) None of the above

Question 10:

```
def wilijiliti(n : Int) : Int = {
  def loop (acc : Int, n : Int) : Int = {
     if (n==0) acc
     else if (n%5 == 0) loop (acc+1, n/10)
         else loop (acc, n/10)
  }
  loop (0,n)
}
```

What does wilijiliti do?

1) checks if the last digit in a number is 0 or 5
2) checks if the first digit in a number is 0
3) checks if the last digit in a number is 5
4) counts the number of digits in a number that are 0 or 5

Question 11: def test(x : Int, y : Int) : Int = y * y For the function call **test(2,3)** determine which evaluation strategy is the fastest (takes the least number of steps)?
NOTE: in Scala def f (n : Int) : Int is same as in Java int f (int n)

1) Call by value faster
2) Call by name faster
3) Same number of steps
4) Evaluations do not terminate

Question 12: def test(x : Int, y : Int) : Int = y * y For the function call **test(2,3+4)** determine which evaluation strategy is the fastest (takes the least number of steps)?
NOTE: in Scala def f (n : Int) : Int is same as in Java int f (int n)

1) Call by value faster
2) Call by name faster
3) Same number of steps
4) Evaluations do not terminate

Question 13: def test(x : Int, y : Int) : Int = y * y For the function call **test(2+3,3)** determine which evaluation strategy is the fastest (takes the least number of steps)?
NOTE: in Scala def f (n : Int) : Int is same as in Java int f (int n)

1) Call by value faster
2) Call by name faster
3) Same number of steps

4) Evaluations do not terminate

Question 14: def test(x : Int, y : Int) : Int = y * y For the function call **test(2+3,3+4)** determine which evaluation strategy is the fastest (takes the least number of steps)?
NOTE: in Scala def f (n : Int) : Int is same as in Java int f (int n)

1) Call by value faster
2) Call by name faster
3) Same number of steps
4) Evaluations do not terminate

Question 15: def test(x : Int, y : Int) : Int = x + y For the function call **test(2+3,3+4)** determine which of the following expressions is the first step of evaluation using call by name?
NOTE: in Scala def f (n : Int) : Int is same as in Java int f (int n)

1) (2 + 3) + (3 + 4)
2) 5 + 7
3) test (5, 4 + 5)
4) None of these

Question 16: def test(x : Int, y : Int) : Int = x + y For the function call **test(2+3,3+4)** determine which of the following expressions is the first step of evaluation using call by name?
NOTE: in Scala def f (n : Int) : Int is same as in Java int f (int n)

1) (2 + 3) + (3 + 4)
2) 5 + 7
3) test (5, 4 + 5)
4) None of these

Question 17: def test(x : Int, y : Int) : Int = x + y For the function call **test(2+3,4)** determine which of the following expressions is the second step of evaluation using call by name?
NOTE: in Scala def f (n : Int) : Int is same as in Java int f (int n)

1) (2 + 3) + 4
2) test (5 , 4)
3) 5 + 4
4) None of these

Question 18: def test(x : Int, y : Int) : Int = x + y For the function call **test(2,3+4)** determine which of the following expressions is the second step of evaluation using call by name?
NOTE: in Scala def f (n : Int) : Int is same as in Java int f (int n)

1) 2 + (3 + 4)
2) test (2 , 7)
3) 2 + 7
4) None of these

Question 19: def test(x : Int, y : Int, z : Int) : Int = y * z For the function call **test(2, 3+4, 5+6)** the first step of evaluation is test(2, 7, 5+6) , which evaluation strategy will result in this step? NOTE: in Scala def f (n : Int) : Int is same as in Java int f (int n)

1) Call by value
2) Call by name
3) Both result in same step
4) Evaluations do not terminate

Question 20: def test(x : Int, y : Int, z : Int) : Int = y * z For the function call **test(2, 3+4, 5+6)** the first step of evaluation is (3 + 4) * (5 + 6) , which evaluation strategy will result in this step? NOTE: in Scala def f (n : Int) : Int is same as in Java int f (int n)

1) Call by value
2) Call by name
3) Both result in same step
4) Evaluations do not terminate

D Textual pretest used in the dual eye-tracking study for MOOCs

Instructions: Please answer the questions you are sure about. Please do not make random guesses.

Question 1. The membrane potential of the cell is constant.
1) True *2)* False

Question 2. The original cause of the resting potential is the fact that the amount of the positive ions which diffuse to the interior is slightly more than the amount of the positive ions which diffuse to the exterior.
1) True *2)* False

Question 3. The original cause of the resting potential is the fact that the potassium ions diffuse faster than sodium ions.
1) True *2)* False

Question 4. Sodium-Potassium pump brings the sodium ions in and potassium ions are expelled through the membrane.
1) True *2)* False

Question 5. Which of the following phenomena explains that the resting potential is negative?

(a) There are more negative ions than positive ions in the liquid that is in the interior of the neuron.
1) True *2)* False

(b) The negative ions that diffuse into the interior of the neuron are more than those which diffuse outward.
1) True *2)* False

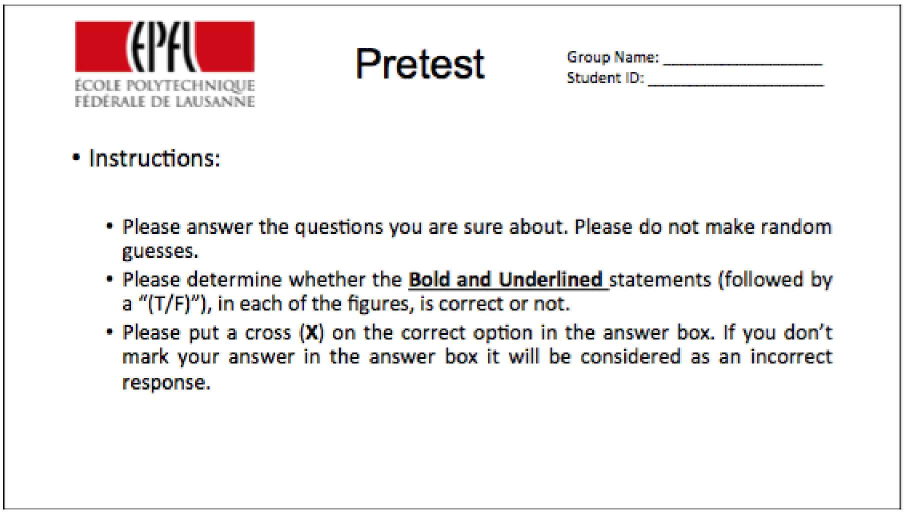
Question 6. What would happen if the sodium-potassium pump were artificially blocked?

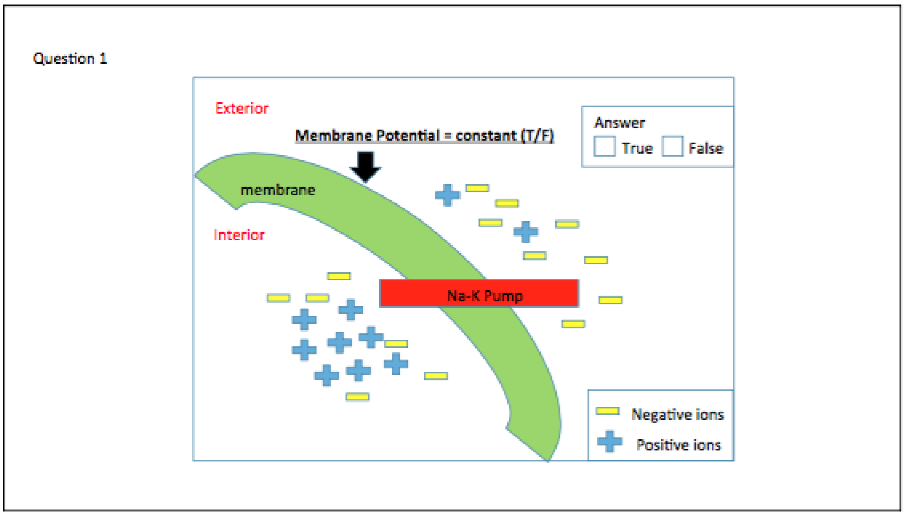
(a) This would lead to the disappearance of the concentration gradients of K+ and Na+ ions on either side of the membrane.
1) True *2)* False

(b) Many potassium ions would accumulate in the interior of the neuron and the neuron no longer works.
1) True *2)* False

Question 7. The diffusion of positive ions to the outside is faster than the diffusion of positive ions to the inside of the neuron.
1) True *2)* False

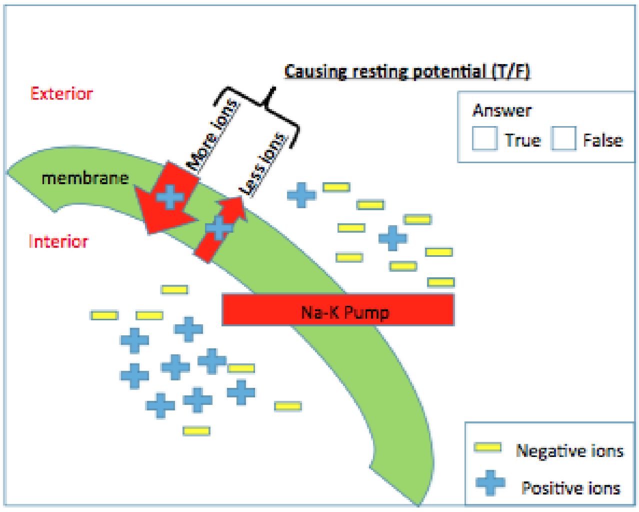
E Schema based pretest used in the dual eye-tracking study for MOOCs



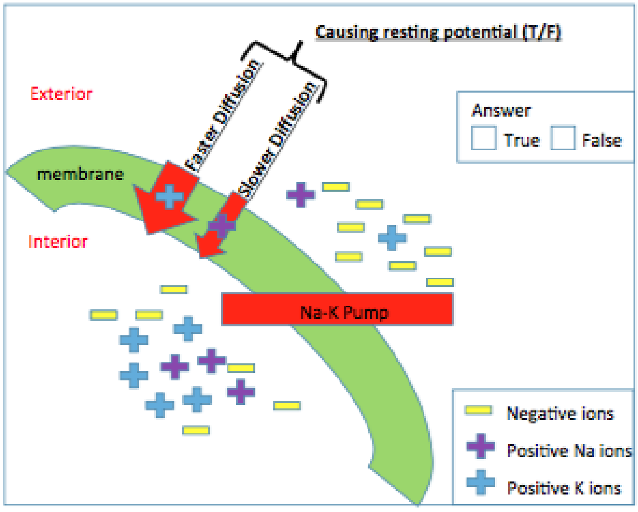


Appendix E. Schema based pretest used in the dual eye-tracking study for MOOCs

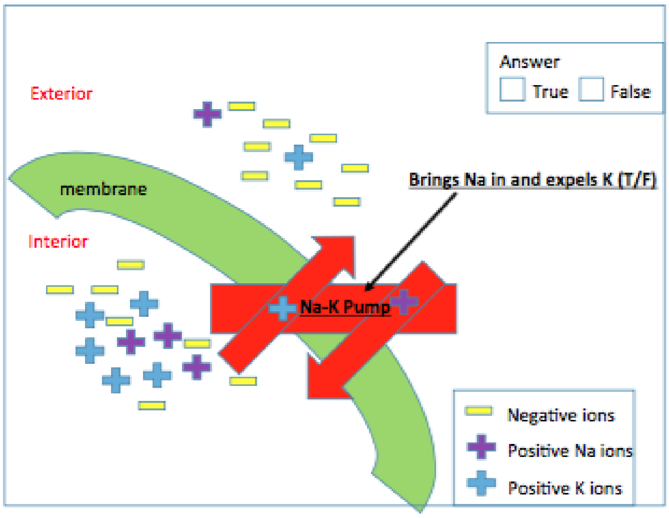
Question 2



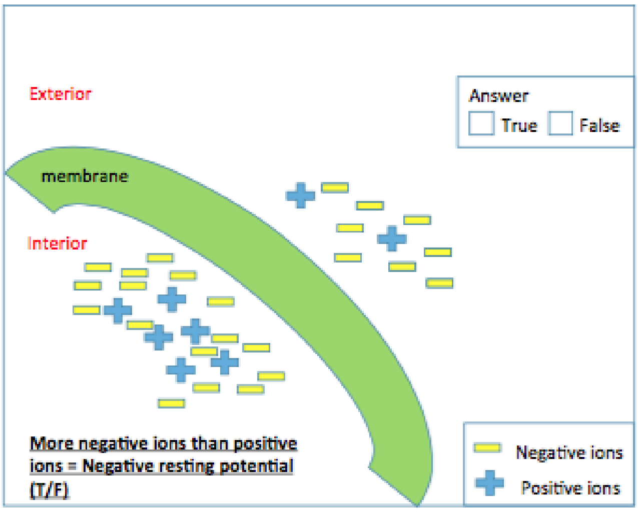
Question 3



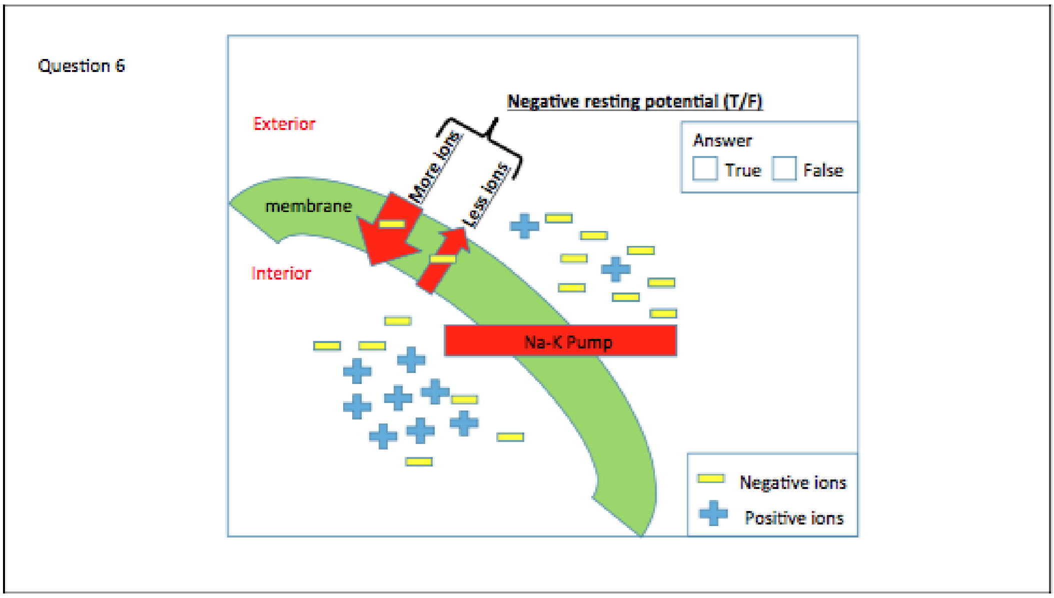
Question 4

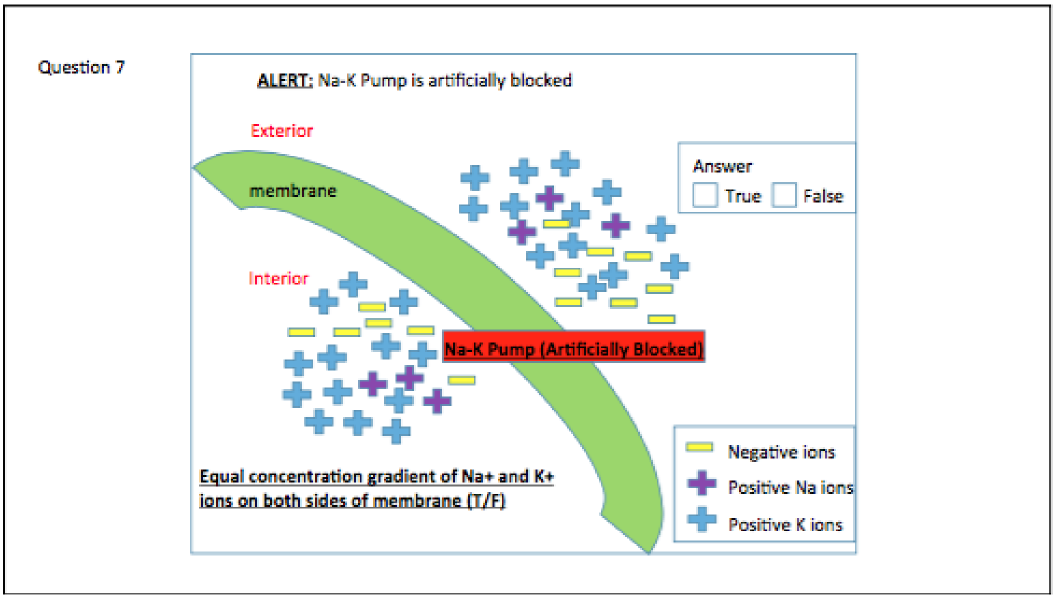


Question 5

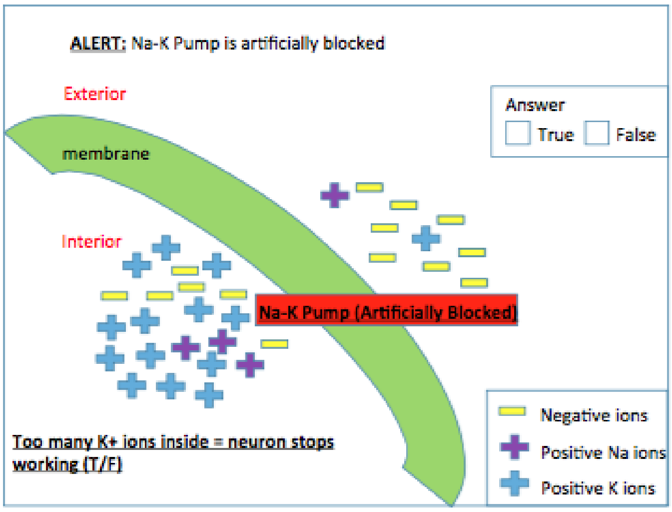


Appendix E. Schema based pretest used in the dual eye-tracking study for MOOCs

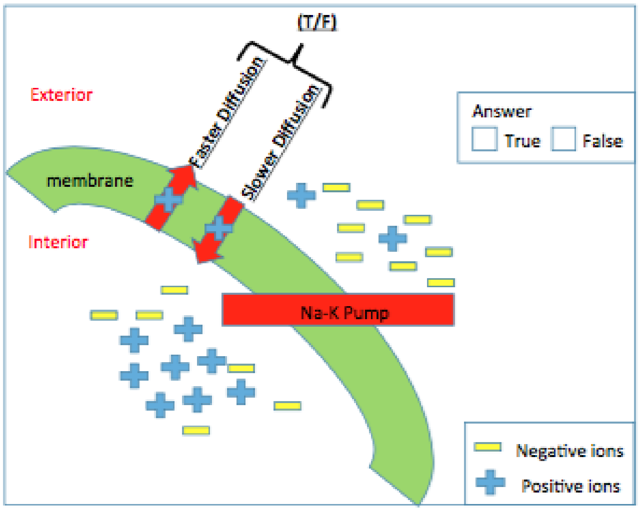




Question 8



Question 9



F Posttest used in the dual eye-tracking study for MOOCs

Instructions: Please answer the questions you are sure about. Please do not make random guesses.

Question 1. The higher the concentration of Na+ ions in the interior of the cell is, the more positive the resting potential is.
1) True *2)* False

Question 2. The most important cause of the resting potential is the fact that Na+ channels are highly permeable for Na+ and let many Na+ ions diffuse inside.
1) True *2)* False

Question 3. When the membrane is at resting potential sodium ions are attracted towards interior of the neuron due to an electrical gradient and a concentration gradient.
1) True *2)* False

Question 4. If the membrane was permeable only to sodium, assuming normal concentrations of ions inside and outside the membrane, the resting potential would be about +50mV.
1) True *2)* False

Question 5. What would happen if the sodium-potassium pump is artificially blocked?

(a) The membrane would have a more positive potential than normal rest.
1) True *2)* False

(b) This would lead to a decrease in membrane potential between the inside and outside areas of the neuron.
1) True *2)* False

Question 6. The electric potential is equal to zero as long as the recording electrode is positioned outside of the membrane of the neuron.

1) True *2)* False

Question 7. At rest, the positive ions are attracted by the charges outside the membrane and the negative ions are attracted by the charges inside the membrane.
1) True *2)* False

Question 8. The sodium-potassium pump pumps the sodium and potassium ions in the same direction as the concentration gradient.
1) True *2)* False

Question 9. The higher the concentration of K+ ions outside of the neuron is, the more negative the resting potential is, all other conditions being equal.
1) True *2)* False

Bibliography

Bruce Abernethy and David G Russell. The relationship between expertise and visual search strategy in a racquet sport. *Human movement science*, 6(4):283–319, 1987.

Serkan Alkan and Kursat Cagiltay. Studying computer game learning experience through eye tracking. *British Journal of Educational Technology*, 38(3):538–542, 2007.

Paul D Allopenna, James S Magnuson, and Michael K Tanenhaus. Tracking the time course of spoken word recognition using eye movements: Evidence for continuous mapping models. *Journal of memory and language*, 38(4):419–439, 1998.

Franck Amadieu, Tamara Van Gog, Fred Paas, André Tricot, and Claudette Mariné. Effects of prior knowledge and concept-map structure on disorientation, cognitive load, and learning. *Learning and Instruction*, 19(5):376–386, 2009.

John R Anderson. Spanning seven orders of magnitude: A challenge for cognitive modeling. *Cognitive Science*, 26(1):85–112, 2002.

Prashant Baheti, Laurie Williams, Edward Gehringer, and David Stotts. Exploring pair programming in distributed object-oriented team projects. In *Educator's Workshop, OOPSLA*, pages 4–8. Citeseer, 2002.

Dana H Ballard, Mary M Hayhoe, Feng Li, Steven D Whitehead, JP Frisby, JG Taylor, and RB Fisher. Hand-eye coordination during sequential tasks [and discussion]. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 337(1281):331–339, 1992.

Roman Bednarik and Markku Tukiainen. An eye-tracking methodology for characterizing program comprehension processes. In *Proceedings of the 2006 symposium on Eye tracking research & applications*, pages 125–132. ACM, 2006.

Roman Bednarik, Niko Myller, Erkki Sutinen, and Markku Tukiainen. Program visualization: Comparing eye-tracking patterns with comprehension summaries and performance. In *Proceedings of the 18th Annual Psychology of Programming Workshop*, pages 66–82, 2006.

Ted J Biggerstaff, Bharat G Mitbander, and Dallas E Webster. Program understanding and the concept assignment problem. *Communications of the ACM*, 37(5):72–82, 1994.

John Biggs, David Kember, and Doris YP Leung. The revised two-factor study process questionnaire: R-spq-2f. *British Journal of Educational Psychology*, 71(1):133–149, 2001.

Jeffrey Bonar and Elliot Soloway. Uncovering principles of novice programming. In *Proceedings of the 10th ACM SIGACT-SIGPLAN symposium on Principles of programming languages*, pages 10–13. ACM, 1983.

Susan E Brennan, Xin Chen, Christopher A Dickinson, Mark B Neider, and Gregory J Zelinsky. Coordinating cognition: The costs and benefits of shared gaze during collaborative search. *Cognition*, 106(3):1465–1477, 2008.

Ruven Brooks. Towards a theory of the comprehension of computer programs. *International journal of man-machine studies*, 18(6):543–554, 1983.

Georg Buscher, Andreas Dengel, and Ludger van Elst. Query expansion using gaze-based feedback on the subdocument level. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 387–394. ACM, 2008.

Brian Byrne, Peter Freebody, and Anne Gates. Longitudinal data on the relations of word-reading strategies to comprehension, reading time, and phonemic awareness. *Reading Research Quarterly*, pages 141–151, 1992.

Neil Charness, Eyal M Reingold, Marc Pomplun, and Dave M Stampe. The perceptual aspect of skilled performance in chess: Evidence from eye movements. *Memory & cognition*, 29(8): 1146–1152, 2001.

William G Chase and Herbert A Simon. Perception in chess. *Cognitive psychology*, 4(1):55–81, 1973.

Mauro Cherubini and Pierre Dillenbourg. The effects of explicit referencing in distance problem solving over shared maps. In *Proceedings of the 2007 international ACM conference on Supporting group work*, pages 331–340. ACM, 2007.

Andrew SA Chetwood, Ka-Wai Kwok, Loi-Wah Sun, George P Mylonas, James Clark, Ara Darzi, and Guang-Zhong Yang. Collaborative eye tracking: a potential training tool in laparoscopic surgery. *Surgical endoscopy*, 26(7):2003–2009, 2012.

Rick Dale, Natasha Z Kirkham, and Daniel C Richardson. The dynamics of reference and shared visual attention. *Frontiers in psychology*, 2, 2011.

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

Sidney D'Mello, Andrew Olney, Claire Williams, and Patrick Hays. Gaze tutor: A gaze-reactive intelligent tutoring system. *International Journal of human-computer studies*, 70(5):377–398, 2012.

Sarah Lynn Dowhower. Effects of repeated reading on second-grade transitional readers' fluency and comprehension. *Reading Research Quarterly*, pages 389–406, 1987.

Andrew T. Duchowski, Nathan Cournia, Brian Cumming, Daniel McCallum, Anand Gramopadhye, Joel Greenstein, Sajay Sadasivan, and Richard A. Tyrrell. Visual deictic reference in a collaborative virtual environment. In *Proceedings of the 2004 symposium on Eye tracking research & applications*, ETRA '04, New York, NY, USA, 2004. ACM. ISBN 1-58113-825-3. doi: 10.1145/968363.968369. URL http://doi.acm.org/10.1145/968363.968369.

Darren Gergle and Alan T Clark. See what i'm saying?: using dyadic mobile eye tracking to study collaborative reference. In *Proceedings of the ACM 2011 conference on Computer supported cooperative work*, pages 435–444. ACM, 2011.

Lewis R Goldberg. A broad-bandwidth, public domain, personality inventory measuring the lower-level facets of several five-factor models. *Personality psychology in Europe*, 7:7–28, 1999.

Judith Good and Paul Brna. Toward authentic measures of program comprehension. In *Proceedings of the Fifteenth Annual Workshop of the Psychology of Programming Interest Group (PPIG 2003)*, pages 29–49. Citeseer, 2003.

Judith Good and Paul Brna. Program comprehension and authentic measurement:: a scheme for analysing descriptions of programs. *International Journal of Human-Computer Studies*, 61(2):169–185, 2004.

John Mordechai Gottman and Anup Kumar Roy. *Sequential Analysis: A Guide for Behavorial Researchers*. Cambridge University Press, 1990.

Elizabeth R Grant and Michael J Spivey. Eye movements and problem solving guiding attention guides thought. *Psychological Science*, 14(5):462–466, 2003.

Zenzi M Griffin and Kathryn Bock. What the eyes say about speaking. *Psychological science*, 11 (4):274–279, 2000.

Joy E Hanna and Susan E Brennan. Speakers? eye gaze disambiguates referring expressions early during face-to-face conversation. *Journal of Memory and Language*, 57(4):596–615, 2007.

Joanne L Harbluk, Y Ian Noy, Patricia L Trbovich, and Moshe Eizenman. An on-road assessment of cognitive distraction: Impacts on drivers? visual behavior and braking performance. *Accident Analysis & Prevention*, 39(2):372–379, 2007.

Mary Hegarty, Richard E Mayer, and Carolyn E Green. Comprehension of arithmetic word problems: Evidence from students' eye fixations. *Journal of Educational Psychology*, 84(1): 76, 1992.

Prateek Hejmady and N. Hari Narayanan. Visual attention patterns during program debugging with an ide. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1221-9. doi: 10.1145/2168556.2168592. URL http://doi.acm.org/10.1145/2168556.2168592.

Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka, and Joost Van de Weijer. *Eye tracking: A comprehensive guide to methods and measures.* Oxford University Press, 2011.

Natasha Jaques, Cristina Conati, Jason M Harley, and Roger Azevedo. Predicting affect from gaze data during interaction with an intelligent tutoring system. In *Intelligent Tutoring Systems*, pages 29–38. Springer, 2014.

Patrick Jermann. Computer support for interaction regulation in collaborative problem-solving. *Unpublished Ph. D. thesis, University of Geneva, Switzerland*, 2004.

Patrick Jermann and Marc-Antoine Nüssli. Effects of sharing text selections on gaze cross-recurrence and interaction quality in a pair programming task. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, pages 1125–1134. ACM, 2012.

Patrick Jermann, Marc-Antoine Nüssli, and Weifeng Li. Using dual eye-tracking to unveil coordination and expertise in collaborative tetris. In *Proceedings of the 24th BCS Interaction Specialist Group Conference*, pages 36–44. British Computer Society, 2010.

W Lewis Johnson and Elliot Soloway. Proust: Knowledge-based program understanding. *Software Engineering, IEEE Transactions on*, pages 267–275, 1985.

Gary Jones. Testing two cognitive theories of insight. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 29(5):1017, 2003.

Marcel A Just and Patricia A Carpenter. A theory of reading: from eye fixations to comprehension. *Psychological review*, 87(4):329, 1980.

Marcel Adam Just and Patricia A Carpenter. Eye fixations and cognitive processes. *Cognitive psychology*, 8(4):441–480, 1976.

Christoph P Kaller, Benjamin Rahm, Kristina Bolkenius, and Josef M Unterrainer. Eye movements and visuospatial problem solving: Identifying separable phases of complex cognition. *Psychophysiology*, 46(4):818–830, 2009.

Günther Knoblich, Stellan Ohlsson, Hilde Haider, and Detlef Rhenius. Constraint relaxation and chunk decomposition in insight problem solving. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 25(6):1534, 1999.

Günther Knoblich, Stellan Ohlsson, and Gary E Raney. An eye movement study of insight problem solving. *Memory & Cognition*, 29(7):1000–1009, 2001.

Jürgen Koenemann and Scott P Robertson. Expert problem solving strategies for program comprehension. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 125–130. ACM, 1991.

Janet L Kolodner. Towards an understanding of the role of experience in the evolution from novice to expert. *International Journal of Man-Machine Studies*, 19(5):497–518, 1983.

Stanley Letovsky. Cognitive processes in program comprehension. *Journal of Systems and software*, 7(4):325–339, 1987.

Nan Li, Łukasz Kidzinski, Patrick Jermann, and Pierre Dillenbourg. How do in-video interactions reflect perceived video difficulty. In *In EMOOCs 2015, the third MOOC European Stakeholders Summit,*, 2015.

Damien Litchfield and Linden J Ball. Using another's gaze as an explicit aid to insight problem solving. *The Quarterly Journal of Experimental Psychology*, 64(4):649–656, 2011.

Yan Liu, Pei-Yun Hsueh, Jennifer Lai, Mirweis Sangin, M-A Nussli, and Pierre Dillenbourg. Who is the expert? analyzing gaze data to predict expertise level in collaborative applications. In *Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on*, pages 898–901. IEEE, 2009.

Robert G Lord and Paul E Levy. Moving from cognition to action: A control theory perspective. *Applied Psychology*, 43(3):335–367, 1994.

James N MacGregor, Thomas C Ormerod, and Edward P Chronicle. Information processing and insight: A process model of performance on the nine-dot and related problems. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 27(1):176, 2001.

Richard E Mayer. Unique contributions of eye-tracking research to the study of learning with graphics. *Learning and instruction*, 20(2):167–171, 2010.

Antje S Meyer, Astrid M Sleiderink, and Willem JM Levelt. Viewing and naming objects: Eye movements during noun phrase production. *Cognition*, 66(2):B25–B33, 1998.

Keith K Millis and Anne King. Rereading strategically: The influences of comprehension ability and a prior reading on the memory for expository text. *Reading Psychology*, 22(1):41–65, 2001.

Allen Newell. *Unified theories of cognition*. Harvard University Press, 1994.

Marc-Antoine Nüssli. Dual eye-tracking methods for the study of remote collaborative problem solving. 2011.

Marc-Antoine Nüssli, Patrick Jermann, Mirweis Sangin, and Pierre Dillenbourg. Collaboration and abstract representations: towards predictive models based on raw speech and eye-tracking data. In *Proceedings of the 9th international conference on Computer supported collaborative learning-Volume 1*, pages 78–82. International Society of the Learning Sciences, 2009.

Alice Oh, Harold Fox, Max Van Kleek, Aaron Adler, Krzysztof Gajos, Louis-Philippe Morency, and Trevor Darrell. Evaluating look-to-talk: a gaze-aware interface in a collaborative environment. In *CHI'02 Extended Abstracts on Human Factors in Computing Systems*, pages 650–651. ACM, 2002.

Erol Ozcelik, Turkan Karakus, Engin Kursun, and Kursat Cagiltay. An eye-tracking study of how color coding affects multimedia learning. *Computers & Education*, 53(2):445–453, 2009.

Scott G Paris and Janis E Jacobs. The benefits of informed instruction for children's reading awareness and comprehension skills. *Child development*, pages 2083–2093, 1984.

Derrick Parkhurst, Klinton Law, and Ernst Niebur. Modeling the role of salience in the allocation of overt visual attention. *Vision research*, 42(1):107–123, 2002.

Nancy Pennington. Empirical studies of programmers: second workshop. chapter Comprehension strategies in programming. Ablex Publishing Corp., 1987. ISBN 0-89391-461-4. URL http://dl.acm.org/citation.cfm?id=54968.54975.

Sami Pietinen, Roman Bednarik, Tatiana Glotova, Vesa Tenhunen, and Markku Tukiainen. A method to study visual attention aspects of collaboration: eye-tracking pair programmers simultaneously. In *Proceedings of the 2008 symposium on Eye tracking research & applications*, pages 39–42. ACM, 2008.

Sami Pietinen, Roman Bednarik, and Markku Tukiainen. Shared visual attention in collaborative programming: a descriptive analysis. In *proceedings of the 2010 ICSE workshop on cooperative and human aspects of software engineering*, pages 21–24. ACM, 2010.

Zahar Prasov and Joyce Y Chai. What's in a gaze?: the role of eye-gaze in reference resolution in multimodal conversational interfaces. In *Proceedings of the 13th international conference on Intelligent user interfaces*, pages 20–29. ACM, 2008.

Helmut Prendinger, Tobias Eichner, Elisabeth André, and Mitsuru Ishizuka. Gaze-based infotainment agents. In *Proceedings of the international conference on Advances in computer entertainment technology*, pages 87–90. ACM, 2007.

Eyal M Reingold, Neil Charness, Marc Pomplun, and Dave M Stampe. Visual span in expert chess players: Evidence from eye movements. *Psychological Science*, 12(1):48–55, 2001.

David Reinking. Computer-mediated text and comprehension differences: The role of reading time, reader preference, and estimation of learning. *Reading Research Quarterly*, pages 484–498, 1988.

Daniel C Richardson and Rick Dale. Looking to understand: The coupling between speakers' and listeners' eye movements and its relationship to discourse comprehension. *Cognitive science*, 29(6):1045–1060, 2005.

Daniel C Richardson, Rick Dale, and Natasha Z Kirkham. The art of conversation is coordination common ground and the coupling of eye movements during dialogue. *Psychological science*, 18(5):407–413, 2007.

Hubert Ripoll, Yves Kerlirzin, Jean-François Stein, and Bruno Reine. Analysis of information processing, decision making, and visual strategies in complex problem solving sport situations. *Human Movement Science*, 14(3):325–349, 1995.

Pablo Romero, Rudi Lutz, Richard Cox, and Benedict Du Boulay. Co-ordination of multiple external representations during java program debugging. In *Human Centric Computing Languages and Environments, 2002. Proceedings. IEEE 2002 Symposia on*, pages 207–214. IEEE, 2002.

Mirweis Sangin, Gaëlle Molinari, Marc-Antoine Nüssli, and Pierre Dillenbourg. How learners use awareness cues about their peer's knowledge?: insights from synchronized eye-tracking data. In *Proceedings of the 8th international conference on International conference for the learning sciences-Volume 2*, pages 287–294. International Society of the Learning Sciences, 2008.

Bonita Sharif, Michael Falcone, and Jonathan I. Maletic. An eye-tracking study on the role of scan time in finding source code defects. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1221-9. doi: 10.1145/2168556.2168642. URL http://doi.acm.org/10.1145/2168556.2168642.

Ben Shneiderman and Richard Mayer. Syntactic/semantic interactions in programmer behavior: A model and experimental results. *International Journal of Computer & Information Sciences*, 8(3):219–238, 1979.

David A Slykhuis, Eric N Wiebe, and Len A Annetta. Eye-tracking students' attention to powerpoint photographs in a science education setting. *Journal of Science Education and Technology*, 14(5-6):509–520, 2005.

Elliot Soloway and Kate Ehrlich. Empirical studies of programming knowledge. *Software Engineering, IEEE Transactions on*, (5):595–609, 1984.

Randy Stein and Susan E Brennan. Another person's eye gaze as a cue in solving programming problems. In *Proceedings of the 6th international conference on Multimodal interfaces*, pages 9–15. ACM, 2004.

Kar-Han Tan, Ian Robinson, Ramin Samadani, Bowon Lee, Dan Gelb, Alex Vorbau, Bruce Culbertson, and John Apostolopoulos. Connectboard: A remote collaboration system that supports gaze-aware interaction and sharing. In *Multimedia Signal Processing, 2009. MMSP'09. IEEE International Workshop on*, pages 1–6. IEEE, 2009.

Catherine Thevenot and Jane Oakhill. A generalization of the representational change theory from insight to non-insight problems: The case of arithmetic word problems. *Acta psychologica*, 129(3):315–324, 2008.

Laura E Thomas and Alejandro Lleras. Moving eyes and moving thought: On the spatial compatibility between eye movements and cognition. *Psychonomic bulletin & review*, 14(4): 663–668, 2007.

Scott R Tilley, Dennis B Smith, and Santanu Paul. Towards a framework for program understanding. In *wpc*, page 19. IEEE, 1996.

Alexander Toet. Gaze directed displays as an enabling technology for attention aware systems. *Computers in Human Behavior*, 22(4):615–647, 2006.

Roland Tormey and Ingrid LeDuc. The activating student knowledge (ask) method in lectures. In *In Proceedings of Proceedings of Educational Development in a changing world 2014.*, 2014.

Tamara Van Gog, Fred Paas, and Jeroen Van Merriënboer. Uncovering expertise-related differences in troubleshooting performance: Combining eye movement and concurrent verbal protocol data. 2005a.

Tamara Van Gog, Fred Paas, Jeroen JG van Merriënboer, and Puk Witte. Uncovering the problem-solving process: cued retrospective reporting versus concurrent and retrospective reporting. *Journal of Experimental Psychology: Applied*, 11(4):237, 2005b.

Tamara Van Gog, Halszka Jarodzka, Katharina Scheiter, Peter Gerjets, and Fred Paas. Attention guidance during example study via the model's eye movements. *Computers in Human Behavior*, 25(3):785–791, 2009.

Roel Vertegaal, Ivo Weevers, and Changuk Sohn. Gaze-2: An attentive video conferencing system. In *CHI'02 Extended Abstracts on Human Factors in Computing Systems*, pages 736–737. ACM, 2002.

Anneliese Von Mayrhauser et al. Program comprehension during software maintenance and evolution. *Computer*, 28(8):44–55, 1995.

Hua Wang, Mark Chignell, and Mitsuru Ishizuka. Empathic tutoring software agents using real-time eye tracking. In *Proceedings of the 2006 symposium on Eye tracking research & applications*, pages 73–78. ACM, 2006.

Laurie A. Williams and Robert R. Kessler. All I really need to know about pair programming I learned in kindergarten. *Commun. ACM*, 43(5):108–114, May 2000. ISSN 0001-0782. doi: 10.1145/332833.332848. URL http://dx.doi.org/10.1145/332833.332848.

Gregory J Zelinsky and Gregory L Murphy. Synchronizing visual and language processing: An effect of object name length on eye movements. *Psychological Science*, 11(2):125–131, 2000.

Kshitij Sharma

Rue de Lausanne, 19
CH-1020 Renens
+41 78 714 84 38
kshitij.sharma@epfl.ch

Education

2011–2015 **PhD Student in Computer Science**, *École Polytechnique Fédérale de Lausanne*, Switzerland.
Title: Gaze Analysis methods for Learning Analytics

2009–2011 **Masters in Information Technology**, *Indian Institute of Information Technology*, Allahabad, India.
Specialization in Intelligent Systems

2005–2009 **Bachelor of Technology**, *Uttar Pradesh Technical University*, Allahabad,Uttar Pradesh, India.
Information Technology

Research Experience

2013–2015 **Eye-tracking MOOC students**, *Swiss National Science Foundation grants CR12I1_132996 and 206021_144975*.

- An eye-tracking study to find the relation between students' gaze patterns and their learning outcome. Considering the teacher-student pair as a collaborating dyad.
- A dual eye-tracking study to find how priming effects the gaze patterns of students in MOOCs and what is the relation between students' individual and their collaborative gaze patterns.
- A system to augment MOOC videos with teachers' gaze.
- A system to provide feedback to students based on their own gaze behaviour.

2011–2013 **Dual eye-tracking with pair programming**, *Swiss National Science Foundation grant CR12I1_132996*.
Analysis of gaze data of a pair collaboratively trying to understand a JAVA program. The main focus was to analyse the quality and effectiveness of collaboration based on the different gaze behaviour of pairs.

2014–2015 **Classroom orchestration load and eye-tracking**.
Tracking teacher's orchestration load in face-to-face classroom situations using eye-tracking.

2013–2014 **Robotics and eye-tracking**.

- A mobile eye-tracking study to find the relation between students' gaze patterns and their understanding of a robot's functionality.
- An eye-tracking study to find how the cognitive context of human robot interaction can effect the gaze patterns of an external observer.

Spring 2014 **MOOC analytics**.
A categorisation scheme for MOOC students to study how does timing and pattern of students' activities affect their engagement.

Fall 2013 **Tangibles and eye-tracking**.
A dual eye-tracking study with mobile eye-trackers to find out the differences in participants' gaze patterns across different tasks and across paper and tangible interfaces.

Fall 2012 **Dual eye-tracking with collaborative gaming**.
A dual eye-tracking experiment in collaboration with Economics department, University of Lausanne to find the relation between stress and reward in competitive and collaborative two player Tetris.

Teaching Experience

Fall 2014 **Digital Education & Learning Analytics**, *Teaching Assistant*, School of Computer and Communication Sciences, École Polytechnique Fédérale de Lausanne.
Course coordination, project supervision

Fall 2013 **Computer-Supported Collaborative Work**, *Teaching Assistant*, School of Computer and Communication Sciences, École Polytechnique Fédérale de Lausanne.
Course coordination, project supervision, conducting user-studies

Fall 2012 **Introduction to Algorithms**, *Teaching Assistant*, School of Computer and Communication Sciences, École Polytechnique Fédérale de Lausanne.
Course coordination, project supervision

Computer Skills

Languages C#, Java, C++

Statistical Tools R

Others LaTeX, ELAN

Languages

Hindi **Native speaker**

English **Advanced** *Bilingual proficiency*

French **Intermediate** *Average reading and writing skills*

Publications

Book Chapters

2015 K. Sharma, P. Jermann, P. Dillenbourg, *"Dual Eye-tracking"*, Submitted in Handbook of Learning Analytics and Educational Data Mining, 2015.

Journal Articles

2015 K. Sharma, H. Verma, D. Caballero, P. Jermann, P. Dillenbourg, *"Shaping Learners' Attention in Massive Open Online Courses"*, Submitted in International Journal in Higher Education (Special Issue on MOOCs), 2015.

2015 S. Lemaignan, K. Sharma, A. R. Jha, P. Dillenbourg, *"Shaping Learners' Attention in Massive Open Online Courses"*, Submitted in International Journal of Human Robot Interaction, 2015.

Proceedings

2015 K. Sharma, D. Caballero, H. Verma, P. Jermann, P. Dillenbourg, *"Looking AT versus Looking THROUGH: A Dual Eye-Tracking Study in MOOC Context"*, Accepted in Proceedings of 11th International Conference of Computer Supported Collaborative Learning, Gothenburg, Sweden, CSCL, 2015.

2015 K. Sharma, P. Jermann, P. Dillenbourg, *"Identifying Styles and Paths toward success in MOOCs"*, Accepted in Proceedings of 8th International Conference of Educational Data Mining, Madrid, Spain, EDM, 2015.

2015 L. P. Prieto, K. Sharma, Y. Wen, P. Dillenbourg, *"The Burden of Facilitating Collaboration: Towards Estimation of Teacher Orchestration Load using Eye-Tracking Measures"*, Accepted in Proceedings of 11th International Conference of Computer Supported Collaborative Learning, Gothenburg, Sweden, CSCL, 2015.

2015 B. Schneider, K. Sharma, S. Cuendet, G. Zufferey, P. Dillenbourg, R. Pea, *"3D Tangibles Facilitate Joint Visual Attention in Dyads"*, Accepted in Proceedings of 11th International Conference of Computer Supported Collaborative Learning, Gothenburg, Sweden, CSCL, 2015.

2015 K. Sharma, P. Jermann, P. Dillenbourg, *"Displaying Teacher's Gaze in a MOOC: Effects on Students' Video Navigation Patterns"*, Accepted in 10th European Conference On Technology Enhanced Learning, Toledo, Spain, EC-TEL 2015.

2015 L. P. Prieto, K. Sharma, P. Dillenbourg, *"Studying Teacher Orchestration Load in Technology-Enhanced Classrooms: A Mixed-method Approach and Case Study"*, Accepted in 10th European Conference On Technology Enhanced Learning, Toledo, Spain, EC-TEL 2015.

2014 K. Sharma, P. Jermann, P. Dillenbourg, *"With-me-ness: A gaze measure of students' attention in MOOCs "*, In Proceedings of 11th International Conference of the Learning Sciences, Boulder, Colorado, USA, ICLS, 2014.

2014 K. Sharma, P. Jermann, P. Dillenbourg, *"How students learn using MOOCs: an eye-tracking insight"*, In Proceedings of 2nd European MOOCs stakeholder's summit, Lausanne, Switzerland, EMOOCs, 2014.

2013 K. Sharma, P. Jermann, M-A. Nüssli, P. Dillenbourg, *"Understanding collaborative program comprehension: Interlacing gaze and dialogues "*, In Proceedings of 10th International Conference of Computer Supported Collaborative Learning, Madison, Wisconsin, USA, CSCL, 2013.

2012 K. Sharma, P. Jermann, M-A. Nüssli, P. Dillenbourg, *"Gaze evidence for different activities in program understanding"*, In Proceedings of 24th Conference of Psychology of Programming interest Group, London, UK, PPIG, 2012.

Workshop Papers

2015 L. P. Prieto, H. S. Alavi, K. Sharma, M. Raca and P. Dillenbourg,, *"Wearable-enhanced classroom orchestration"*, Accepted at Envisioning Wearable Enhanced Learning at EC-TEL 2015, Toledo, Spain, WELL, 2015.

2013 K. Sharma, P. Jermann, M-A. Nüssli, P. Dillenbourg, *"Gaze as a proxy for cognition and communication"*, Workshop on Dual Eye-tracking at CSCL 2013, Madison, Wisconsin, USA, DUET, 2013.

2012 P. Jermann, M-A. Nüssli, K. Sharma, *"Attentional episodes and focus"*, Workshop on Dual Eye-tracking at CSCW 2012, Seattle, Washington, USA, DUET, 2012.

Posters

2014 L. P. Prieto, Y. Wen, D. Caballero, K. Sharma, Y. Wen, P. Dillenbourg, *"Studying Teacher Cognitive Load in Multi-tabletop Classrooms Using Mobile Eye-tracking."*, In Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces, Dresden, Germany, ITS 2014.

Invited Presentations

2015 "Looking Through versus Looking At", *Delft Data Science Seminar: Speeding up the online learning curve,TU Delft, Netherlands*.

2013 "Dual Eye-tracking: Lessons Learnt", *EARLI 2013, SIG 6 and SIG 7 invited double-symposium, TU Munich, Germany*.

Academic Responsibilities

Master Projects

2014 Fall **Measuring anthropomorphism towards robots using eye-tracking**, *Ashish Ranjan Jha*, Section of Computer Science.
Master Program, 1^{st} semester, École Polytechnique Fédérale de Lausanne

2013 Fall **Eye-tracking and robotics**, *Lukas Oliver Hostettler*, Section of Microtechnics.
Master Program, 1^{st} semester, École Polytechnique Fédérale de Lausanne

Referees

Prof. Pierre Dillenbourg, *Computer Human Interaction in Learning and Instruction*, École Polytechnique Fédérale de Lausanne, email: pierre.dillenbourg@epfl.ch.

Dr. Patrick Jermann, *Center for Digital Education*, École Polytechnique Fédérale de Lausanne, email: patrick.jermann@epfl.ch.