An Eye Tracking Study on the Effects of Layout in Understanding the Role of Design Patterns

Bonita Sharif and Jonathan I. Maletic
Department of Computer Science
Kent State University
Kent, Ohio 44242
bsimoes@cs.kent.edu and jmaletic@cs.kent.edu

***Abstract*—The effect of layout in the comprehension of design pattern roles in UML class diagrams is assessed. This work replicates and extends a previous study using questionnaires but uses an eye tracker to gather additional data. The purpose of the replication is to gather more insight into the eye gaze behavior not evident from questionnaire-based methods. Similarities and differences between the studies are presented. Four design patterns are examined in two layout schemes in the context of three open source systems. Fifteen participants answered a series of eight design pattern role detection questions. Results show a significant improvement in role detection accuracy and visual effort with a certain layout for the Strategy and Observer patterns and a significant improvement in role detection time for all four patterns. Eye gaze data indicates classes participating in a design pattern act like visual beacons when they are in close physical proximity and follow the canonical layout, even though they violate some general graph aesthetics.**

Keywords-eye-tracking study; UML class diagram layout; design pattern roles

I. INTRODUCTION

Design patterns [7] are widely used to solve object-oriented design problems. They are also widely taught as part of an undergraduate computer science or software engineering curriculum. Design patterns are typically presented (and taught) using canonical UML class diagram templates [7]. To better understand how the layout of UML class diagrams impacts how we comprehend design pattern usage, we conducted a study [18] using online questionnaires to determine if layout has an effect on how students identify design pattern roles in UML class diagrams. The results of the study indicate a significant improvement in time taken to identify roles in four patterns (Composite, Observer, Strategy, and Singleton) with a certain layout scheme that closely matches the canonical representation. In addition, the Strategy pattern was found to benefit the most from this layout scheme in terms of accuracy and time.

The work presented here replicates this above study using an alternate method of data collection. An eye tracker is used to unobtrusively gather eye gaze data, while subjects are solving tasks. The purpose of replicating this study is to further validate the findings of the previous study using a different sample population as well as gather additional insight into the thought processes via eye gaze data missing from the questionnaire-based study conducted earlier [18]. Moreover, a set of concrete eye-tracking measures representing the visual effort are derived from the eye gaze data that may be used determine the quality of class diagrams. Visual effort is directly linked to the cognitive effort based on the immediacy theory given by Just and Carpenter [11].

In prior work [1, 16, 17], we found layout to have a significant impact on the comprehension of software maintenance tasks. Most UML diagramming tools like MagicDraw and Visual Paradigm try to achieve the best aesthetically looking UML class diagram. However, results from our work show that this is not as important as semantically grouping related classes together. These groups or clusters act as visual beacons analogous to beacons present in source code [3]. When these visual beacons are present in the class diagram they tend to reduce cognitive load and effort needed to solve the task. This premise is validated using quantitative eye-tracking data presented in this study.

The results of this work directly impact both industry and academics. In academia, better layouts can be used to improve the teaching of design patterns. When class diagrams and design patterns are introduced to students, they should also be made aware of how layout can have an impact on comprehension. The focus should be on aesthetics and comprehensibility. In industry, using a good layout will help in accurately identifying the correct solution in less time and effort. If layout reduces the initial effort a software maintainer needs to understand a diagram, more time can be spent on solving the task rather than worrying about correctly tracing a relationship between classes or even failing to see something important due to the nature of the layout. The goal is making the UML class diagram more accessible to the maintainer of the system.

The research questions this paper attempts to address are:

- RQ1: Do clustered layouts improve design pattern role detection accuracy and time in UML class diagrams?
- RQ2: Which design patterns benefit the most from the clustered layouts?
- RQ3: Is there a difference in eye gaze behavior between design experts and novices?

The paper is organized as follows. Section II introduces the two layout schemes used. The experimental setup is given in

Section III. Section IV presents the results. Threats to validity are addressed in Section V followed by the related work and conclusions.

II. CLASS DIAGRAM LAYOUTS

The two layouts used in this experiment are the orthogonal layout and the multi-cluster layout. In each of the layouts, three class stereotypes [2] of control, boundary and entity are visually represented via textual annotations (above the class name) and color. Boundary classes are shown in blue, entity classes are shown in green, and control classes are red. A class fits into one of three stereotypes. Control classes manage interactions between classes. Boundary classes are responsible for communication with users and external entities. Entity classes store persistent data.

The orthogonal layout is based on general aesthetic criteria [4, 6, 9, 13, 14] such as minimizing edge crossings, minimizing edge bends, minimizing edge length, maximizing symmetry, and using 90 degree bends. It does not use information about the class stereotype or semantic meaning in layout positioning. This is a typical layout produced by a commercial tool such as Magic Draw or Visual Paradigm.

The multi-cluster layout is based on forming multiple clusters, where each cluster consists of related classes. Control classes along with their related entity and boundary classes that form a cohesive cluster are grouped closer together. Each cluster has a semantic meaning and is associated to part of a concept/feature in source code or requirements that represents a tightly connected component. This layout depends on the types of relationships that exist between the classes. For example, even though in a generalization hierarchy children are shown immediately below the parent class, in the multi-cluster layout we position the child closer to another class it is associated with or dependent on thus highlighting a particular feature in the system. The number of clusters is usually limited to four or five since each cluster consists of four or five classes on average. In this study, classes participating in a design pattern are shown in one cluster (see Figure 1. for an example of the two layouts).

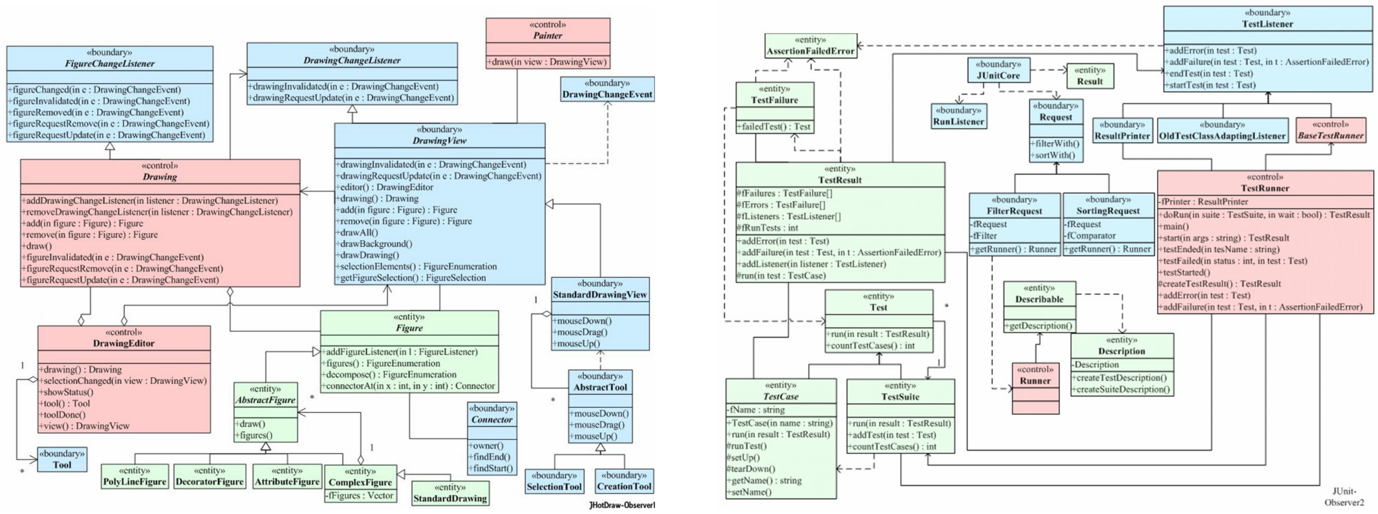
Aesthetic criteria defined in the literature [4, 13, 14] are maintained in both layouts. For example, edge crossings are kept to a minimum and generalizations are drawn to point in one direction. Both the layouts display stereotype information via textual annotations (<<control>>, <<boundary>>, and <<entity>>), as well as color, to control any biases or confounding factors, even though the orthogonal layout does not use the stereotype information. Note that this study does not seek to determine if the presence of stereotype information helped, rather it seeks to determine if the clustered layout helped in identifying roles of a design pattern.

III. EXPERIMENTAL DESIGN

The goal definition template given by Wohlin et al. [21] is used to describe the experiment. The experiment seeks to *analyze* two class diagram layouts *for the purpose of* evaluating their usefulness in understanding design pattern roles *with respect to* effectiveness (accuracy), efficiency (time), and visual effort *from the point of view of the* researcher *in the context of* students and faculty at Kent State University. An overview of the experiment is given in Table I. The main factor being analyzed is the UML class diagram layout. This study also uses a within-subjects design similar to the original study [18]. Each subject answered the role detection questions in four design patterns using both layouts but using different systems to overcome any learning effects. The dependent variables are discussed in Section III.F.

TABLE I. EXPERIMENT OVERVIEW

Goal	Study the effect of two layout schemes for class diagrams in the context of identifying classes and their roles in design patterns using an eye tracker
Main factor	Class diagram layouts with two treatment levels: orthogonal layout, multi-cluster layout
Dependent variables	Accuracy, time, relevance, visual effort



(a) (b)

Figure 1. An example of the multi-cluster (a) layout for JHotDraw and the orthogonal layout (b) for JUnit containing the Observer pattern. For JUnit, TestResult is the Subject and TestListener is the Observer. For JHotDraw, Drawing is the Subject and DrawingView is the concrete Observer with DrawingChangeListener playing the role of the Observer. Both answers are accepted for the Observer role.

A. Eye-Tracking Apparatus

The experiment was conducted using the Tobii 1750 eye tracker (www.tobii.com). It is a video-based remote eye tracker that uses two cameras to capture eye movements. The cameras are built into a 17 inch TFT-LCD screen. The screen resolution was set to 1024 by 768. No head mounting hardware was necessary making the work environment very similar to normal working conditions. The temporal resolution was set at 50 Hz. This eye tracker has a latency of approximately 25-35 ms, and average accuracy is 0.5 degrees which averages to 15 pixels of error. The eye tracker compensates for head movement during the study i.e., the eyes do not have to be focused on the screen all the time.

Analysis was done using the ClearView software that comes with the eye tracker. The study was set up as a double-screen configuration. The first screen is used by the moderator to set up and run the study. The second screen is used by the study subjects to perform the tasks. The moderator was able to get real time feedback of the eye tracking quality during the task. In addition to recording eye gaze data, the Tobii eye tracker also makes an audio/video recording of the study session. The eye gaze data include timestamps, gaze positions, eye positions, pupil size, and validity codes. In this study, we use gaze positions and timestamps to measure visual effort.

B. Hypotheses

The following null hypotheses are formulated based on the research questions. They are based on each of the dependent variables: accuracy, time, and visual effort. The corresponding alternative hypotheses are easily derived from the null hypotheses stating that the multi-cluster layout performs better.

- $H0_A$: There is no significant difference in design pattern role detection *accuracy* between class diagrams in the orthogonal layout and the multi-cluster layout.
- $H0_T$: There is no significant difference in design pattern role detection *time* between class diagrams in the orthogonal layout and the multi-cluster layout.
- $H0_{VE}$: There is no significant difference in the *visual effort* required for design pattern role detection between class diagrams in the orthogonal layout and the multi-cluster layout.

C. Design Patterns, Diagrams, and Subject Systems

The experiment investigates four design patterns: Composite, Observer, Strategy, and Singleton using three open-source systems: JUnit, JHotdraw, and Qt. All of these systems are well designed and use many of the design patterns stated in [7]. See Table II for an overview of the systems used. Since it was not possible to cover all design patterns due to time constraints, we chose the four commonly used ones.

A total of eight diagrams were drawn, two for each design pattern: one in the orthogonal layout and the other in the multi-cluster layout. These diagrams were manually engineered using a UML drawing editor by inspecting the code and online documentation for classes that work closely together towards a specific functional requirement. The classes chosen were part of a module with common functionality. In the case of *Qt*, the design model was first reverse engineered using the *srcTools* framework [20] to identify associations and aggregations. The multi-cluster layout places classes participating in a design pattern in one cluster and was drawn to closely match the canonical form given in [7]. The diagrams had between 17 and 20 classes based on earlier findings [13, 19].

D. Comprehension Task and Stimuli

The main task was to detect the role of classes participating in a design pattern. There were eight questions prepared for this task. See Table III for the questions. The number of crossings and number of classes were kept the same across layouts. The number of relationships were also within the range of ± 3 on average between layouts. Each question was accompanied by a class diagram in one of two layouts. Note that the participants were not asked to identify design patterns in the diagram. They were told that the diagram contains a specific design pattern and were asked to find those classes and assign roles to them. The analysis compares Q1 with Q5, Q2 with Q6, Q3 with Q7 and Q4 with Q8. The diagrams used in this study are referred to as stimuli. An example of a stimulus shown to the subjects is given in Figure 2. excluding the areas of interest marked in red. The participants did not see these areas of interest while solving the tasks.

E. Defining Areas Of Interest

The visual focus of the eyes on a particular location triggers certain mental processes in order to solve a given task [11]. Due to this correlation, visual attention can be used to study the cognitive effort in solving a task. We study visual attention with respect to the areas of interest (AOI) defined below. Figure 2. illustrates the areas of interest used. In this study, we define five areas of interest.

1. Entire stimulus: This involves all the classes and relationships in the diagram. This is characterized by *T* representing total.
2. Design pattern classes: This only involves the classes involved in a particular design pattern. This is represented by *DP*.
3. Design pattern cluster: This area of interest consists of the classes and relationships participating in the design pattern, represented by *DPCluster*.
4. Non-Design pattern classes: These are classes not participating in the design pattern. We represent this with *nonDP*.
5. Task definition: The top left corner of the screen (Figure 2.) consists of the task definition. We do not include eye gaze data in this area in our analysis.

TABLE II. OVERVIEW OF SUBJECT SYSTEMS AND CLASSES USED

System	Domain	Lang.	KLOC	Ver.	#Classes used
JUnit	Testing framework	Java	9	4.6	27
JHotDraw	Drawing Appl.	Java	15	5.1	49
Qt	GUI framework	C++	729	4.3.3	36
Total					112

TABLE III. ROLE DETECTION QUESTIONS

ID	Design Pattern	System	Layout	# crossings	# clusters	# classes	# relations
Q1	Composite	JUnit	Orthogonal	1	4	17	21
Q2	Observer	JUnit	Orthogonal	1	4	20	28
Q3	Strategy	Qt	Multi-cluster	0	-	17	18
Q4	Singleton	Qt	Multi-cluster	0	-	19	19
Q5	Composite	JHotDraw	Multi-cluster	0	-	17	19
Q6	Observer	JHotDraw	Multi-cluster	1	-	20	24
Q7	Strategy	JHotDraw	Orthogonal	1	4	17	20
Q8	Singleton	JHotDraw	Orthogonal	0	4	19	19

F. Dependent Variables

This section discusses the dependent variables used in this study.

Accuracy: The sum of all the scores for each pattern's role assignment. This is an integer number between 0 and maximum number of roles in a pattern.

Time: The amount of time required to detect the roles for each pattern. This is measured in milliseconds as well as seconds.

Relevance: The same as accuracy but ignores the exact matchup of role assignment. The value of this variable may be greater than or equal to the Accuracy variable. When greater, it denotes the mismatch of roles in a pattern.

Visual Effort: The amount of effort needed in terms of eye movements to arrive at the answer. Eight measures are defined to determine the visual effort and are presented next.

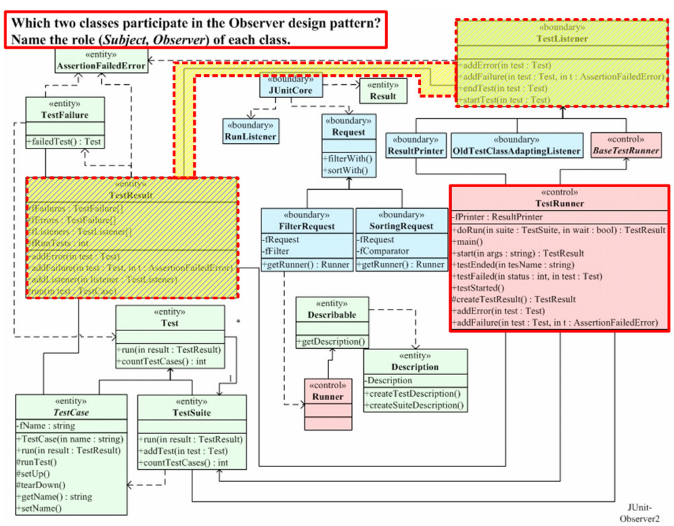


Figure 2. A stimulus with areas of interest overlaid using red rectangles. The top left shows the task definition AOI. Each class's bounding box is an AOI. A design pattern cluster AOI is shown in dotted lines and shaded yellow. The classes at both ends of this cluster form the design pattern classes AOI. All classes outside the design pattern cluster are part of the non design pattern classes AOI.

Two main eye gaze data are eye fixations and saccades. A fixation is the stabilization of the eyes on an object on the stimulus. Saccades are quick movements from fixation to fixation. The eye tracker was set to filter fixations within 20 pixels with duration of at least 40 ms. These settings were chosen based on recommendations given in the eye tracking manual for the specific type of stimuli. The settings were tested prior to the study and were found to be effective in differentiating between objects on the class diagram stimulus. Visual effort is determined using each of the following measures. The parameter for each measure represents the area of interest defined in Section III.E. For each of the first four areas of interest, two visual effort measures are calculated based on the fixation count (1 measure), fixation rate (3 measures), and average fixation duration (4 measures).

- Fixation Count *FC(T)*: The total number of eye fixations on the entire stimulus. This includes all classes and relationships.
- Fixation Rate on Design Pattern Classes *FR(DP)*: The total number of eye fixations on the design pattern classes with respect to all classes on the stimulus.
- Fixation Rate on Design Pattern Cluster *FR(DPCluster)*: The total number of eye fixations on the design pattern classes and relationships with respect to all the classes on the stimulus.
- Fixation Rate on Non-Design Pattern Classes *FR(nonDP)*: The total number of eye fixations on the classes not participating in the design pattern with respect to all classes on the stimulus. These are classes not relevant to the design pattern.
- Average Fixation Duration *AFD(T)*: The average length of time of all fixations in all classes and relationships on the stimulus.
- Average Fixation Duration on Design Pattern Classes *AFD(DP)*: The average length of time of all fixations in classes participating in the design pattern.
- Average Fixation Duration on the Design Pattern Cluster *AFD(DPCluster)*: The average length of time of all fixations on classes and relationships participating in the design pattern.
- Average Fixation Duration on Non-Design Pattern Classes *AFD(nonDP)*: The average length of time of all fixations in classes not participating in the design pattern.

The unit of measure for the average fixation duration is milliseconds. A higher fixation count, duration and fixation rate indicates more effort needed by subjects to solve the task. Each of the above measures is illustrated below.

$$FC(T) = \sum_{a \in \{\text{entire stimulus}\}} f(a) \tag{1}$$

$$FR(DP) = \frac{\sum_{a \in \{\text{DP classes}\}} f(a)}{\sum_{a \in \{\text{DP classes}\} \cup \{\text{nonDP classes}\}} f(a)} \tag{2}$$

$$\mathrm{FR(DPCluster)} = \frac{\sum_{a \in \{\text{DP classes}\} \cup \{\text{DP relationships}\}} f(a)}{\mathrm{FC(T)}} \quad (3)$$

$$\mathrm{FR(nonDP)} = \frac{\sum_{a \in \{\text{nonDP classes}\}} f(a)}{\sum_{a \in \{\text{DP classes}\} \cup \{\text{nonDP classes}\}} f(a)} \quad (4)$$

$$\mathrm{AFD(T)} = \frac{\sum_{a \in \{\text{entire stimulus}\}} g(a)}{\mathrm{FC(T)}} \quad (5)$$

$$\mathrm{AFD(DP)} = \frac{\sum_{a \in \{\text{DP classes}\}} g(a)}{\sum_{a \in \{\text{DP classes}\}} f(a)} \quad (6)$$

$$\mathrm{AFD(DPCluster)} = \frac{\sum_{a \in \{\text{DP classes}\} \cup \{\text{DP relationships}\}} g(a)}{\sum_{a \in \{\text{DP classes}\} \cup \{\text{DP relationships}\}} f(a)} \quad (7)$$

$$\mathrm{AFD(nonDP)} = \frac{\sum_{a \in \{\text{nonDP classes}\}} g(a)}{\sum_{a \in \{\text{nonDP classes}\}} f(a)} \quad (8)$$

where *f(a)* gives the fixation count of an area of interest a, and *g(a)* gives the total gaze time (total time of all fixations) in an area of interest a.

The role detection task is typically solved in two phases. In the first phase, the subject tries to find the classes that participate in the design pattern. This is done through exploring all possible classes to find the relevant ones. In the second phase, they try to determine the roles of classes they suspect to be part of the design pattern. The first is an exploration phase where as the second is more focused.

G. *Participants*

Fifteen volunteers from the Department of Computer Science at Kent State University participated in this study. There were seven undergraduates in their second year of study, six graduate students, and two faculty members. The undergraduates were considered to be novices in design whereas the graduates and faculty members were experts and had more experience in the usage of UML as well as design patterns. The expertise separation was based on a background questionnaire that gathered demographic data on the participants before the study. Two of the subjects were female. All subjects had normal vision. Some wore contact or corrective lenses. The subjects were not aware of the experiment's hypotheses. All of the subjects were first introduced to design patterns in academia. A couple of the experts worked with design patterns in industry. None of them were taught a layout style while learning UML.

H. *Running the Study*

A few days before the study, subjects were given a refresher course on UML class diagrams and design patterns. The study was conducted in a dedicated room accommodating the Tobii eye tracker. The subjects were seated approximately 60 cm away from the screen. The subjects first signed an informed consent form explaining the purpose of the study and procedures used to collect data. They were informed that the purpose was to understand how software engineers interpret design pattern roles in class diagrams. The eye tracker was calibrated using five points on the screen and took approximately one minute. The background color of the calibration was set to white since this was the background of the stimuli used in the study.

The first screen displayed instructions on what the task was. It stated that they were required to identify classes and roles of a particular design pattern. The next four screens, gave a description of each design pattern. They were allowed to study this for as long as they liked. After the subjects understood the goal of the exercise, the actual study, consisting of eight questions, began. The moderator controlled the movement through the tasks to avoid any unnecessary timing delays between subjects. The subjects were asked to verbally state the role of each class in the design pattern. Finally, after all the tasks were completed, a short post-study questionnaire was administered in an interview session by the moderator.

IV. EXPERIMENTAL RESULTS

The accuracy and time analysis is presented first followed by the analysis of eye gaze data. We use the paired-Wilcoxon test for hypotheses testing, due to the within-subjects nature of the study and low sample size.

A. *Accuracy*

The results of the experiment for accuracy are shown in Table IV. See Figure 3. for the descriptive statistics. We find a significant difference in accuracy with the multi-cluster layout for the Strategy and Singleton pattern. The previous study found support for the Strategy pattern but not the Singleton pattern in terms of accuracy.

It is surprising that the Singleton pattern benefits from the multi-cluster layout especially since it consists of only one class. There was one subtle difference in the Singleton pattern diagrams used in this study compared to the original study. A self-dependency relationship was added in a non-singleton class in both the multi-cluster and orthogonal layouts. Even though these classes have a self-dependency, they are not part of the Singleton pattern. The non-singleton class with the self dependency had it's attributes and methods visible in the orthogonal layout but were hidden in the multi-cluster layout. Even though these attributes and methods did not indicate that the class was a singleton, novices tended to choose this class as their answer. We conjecture the Singleton pattern benefitted from the multi-cluster layout due to lower level of detail shown for this non-singleton class which caused subjects especially novices not to choose it as their answer and look elsewhere for another class. This difference in the level of detail between the two layouts was unintentional and was only noticed after the study was conducted. Hence, results on the Singleton pattern need to be considered with caution.

The Observer pattern approaches significance for accuracy (*p-value* = 0.059) and is significant for relevance (*p-value* = 0.045). This shows us that the roles of Subject and Observer were mismatched. In particular, the Observer role in the Observer pattern is detected significantly better in the multi-cluster layout (*p-value* = 0.031). In the post-interview session at the end of the study, subjects stated that Observer was difficult for them to detect. This remark is reflected in the data. Based on the results, we can reject the null hypothesis $H0_A$ for the Strategy pattern. More tests are needed to determine the accuracy of the Singleton pattern with respect to the layout.

B. Time

The multi-cluster layout is significantly better than the orthogonal layout in all four design patterns (See last column of Table IV for *p-value*s) with respect to the time taken to answer the questions. This concurs with the results from the previous study. The effect sizes are much larger in this experiment than in the previous one. A large effect (Cohen's d >= 0.7), i.e., practical significance is noted in the Strategy and Observer patterns. We can reject the null hypothesis $H0_T$ for all four patterns.

C. Visual Effort

We measure visual effort using eight variables, each of which use two main types of eye gaze data: the fixation counts and the average fixation duration. See Table V.

1) Entire Stimulus:

With respect to the entire stimulus, the higher the fixation counts and average fixation durations, the more effort is needed overall. With respect to the Composite and Observer pattern, there are significantly lower fixation counts, FC(T), for the multi-cluster layouts. With respect to AFD(T), we find significantly lower fixation gaze time for the Observer and Strategy patterns. The Composite and Singleton patterns show no difference in average fixation duration across layouts, whereas Strategy and Singleton show no difference in fixation counts across layouts. The measures in this area of interest give an overall indication of visual effort.

2) Design Pattern Classes and Clusters:

The measures in these areas of interest (DP and DPCluster) focus on phase two of the task, where the subject is trying to identify roles after they have identified classes that participate in the design pattern. The fixation rate in design pattern clusters is significantly lower using the multi-cluster layout for the Observer and Strategy patterns (*p-values*= 0.032 and 0.042 respectively). The average fixation duration in design pattern clusters is also significantly lower using the multi-cluster layout for the Observer and Strategy patterns (*p-values*= 0.047 and 0.03 respectively). The Composite and Singleton patterns are not significantly different across layouts. The higher the fixation rate and average fixation duration in the cluster, the more difficult it is to determine the roles. See Figure 4. for box plots on the fixation rate and average fixation duration in the DPCluster AOI.

Considering the design pattern classes only (DP), we find the Observer pattern had a lower fixation rate for the multi-cluster layout and the Strategy pattern had a lower average fixation duration for the multi-cluster layout. No other differences were reported. Since the design pattern cluster covers a larger area than the design pattern classes, more time is spent looking at the cluster than on the design pattern classes themselves. The DP area of interest excludes the fixations on relationship ends. The measures FR(DPCluster) and AFD(DPCluster) give a more accurate picture than the FR(DP) and AFD(DP), since they focus on the classes and relationships. Even though fixations are usually not found on the relationship lines, they are most often found at the relationship ends. These fixations fall outside the bounding box of the class and are not counted in the DP area of interest but are counted in the DPCluster AOI.

3) Non-Design Pattern Classes:

The measures in this area of interest focus on phase one of the task, where the subject is looking for the classes that belong to a particular design pattern. The higher the rate in non-design pattern classes, the more difficult it is to spot the classes in the pattern, i.e., they explored more classes before selecting their answer. The fixation rate for non-design pattern classes, FR(nonDP), shows a significantly higher effort in the orthogonal layout for the Singleton pattern (*p-value* = 0.008). This is also reflected in the accuracy result above. The average fixation duration for non-design pattern classes, AFD(nonDP), is significantly higher in the orthogonal layout for the Composite, Observer, and Singleton patterns (*p-values*=0.002, 0.001, and 0.018 respectively). This implies that trying to search for the relevant classes took much longer in the orthogonal layout than the multi-cluster layout for these patterns.

Based on the results, we can reject the null hypothesis $H0_{VE}$ for the Observer and Strategy patterns, where the multi-cluster layout is shown to reduce visual effort in majority of the AOIs.

TABLE IV. 1-TAILED WILCOXON *P-VALUES* (ALPHA=0.05) FOR ACCURACY, TIME, AND RELEVANCE FOR EACH DESIGN PATTERN. DIRECTIONALITY IMPLIES THAT THE MULTI-CLUSTER LAYOUT IS MORE ACCURATE AND TAKES LESS TIME. COHEN'S D DENOTES THE EFFECT SIZE: 0.2(SMALL), 0.5 (MEDIUM), >=0.8 (LARGE). * INDICATES SIGNIFICANCE.

Role Detection Accuracy				Accuracy (Cohen's d)	Relevance (Cohen's d)	Time (Cohen's d)
Composite Pattern						0.021 * (0.64)
Roles	*Composite*	*Component*	*Leaf*			
	0.250	0.250	0.750	0.188 (0.2)	0.875 (0.06)	
Observer Pattern						0.019 * (0.73)
Roles	*Subject*	*Observer*				
	0.227	0.031 *		0.059 (0.6)	0.045 * (0.7)	
Strategy Pattern						0.036 * (0.70)
Roles	*Concrete Strategy*	*Context*	*Strategy*			
	0.363	0.016 *	0.016 *	0.045 * (0.8)	0.424 (0.1)	
Singleton Pattern						0.024 * (0.40)
Role	*Singleton*					
	0.031 *			0.031 * (0.9)	0.031 * (0.9)	

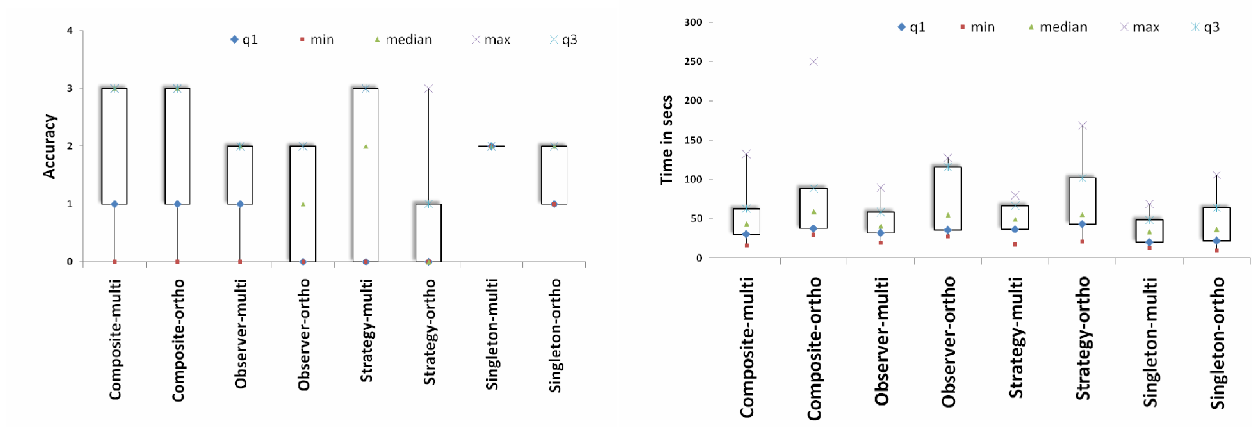


Figure 3. Accuracy and time box plots for patterns across layouts

TABLE V. 1-TAILED WILCOXON P-VALUES (α=0.05) FOR THE VISUAL EFFORT MEASURES. DIRECTIONALITY IMPLIES THE MULTI-CLUSTER LAYOUT HAS A LOWER FIXATION COUNT, RATE AND DURATION.

AOI	Dependent Variable	Composite	Observer	Strategy	Singleton
Entire Diagram	FC(T)	0.014 *	0.042 *	0.126	0.738
	AFD(T)	0.533	0.015 *	0.021 *	0.195
DP classes	FR(DP)	0.076	0.036 *	0.165	0.994
	AFD(DP)	0.165	0.211	0.042 *	0.700
DP cluster	FR(DPCluster)	0.281	0.032 *	0.042 *	0.996
	AFD(DPCluster)	0.281	0.047 *	0.024 *	0.423
Non-DP classes	FR(nonDP)	0.932	0.968	0.849	0.008 *
	AFD(nonDP)	0.002 *	0.001 *	0.874	0.018 *

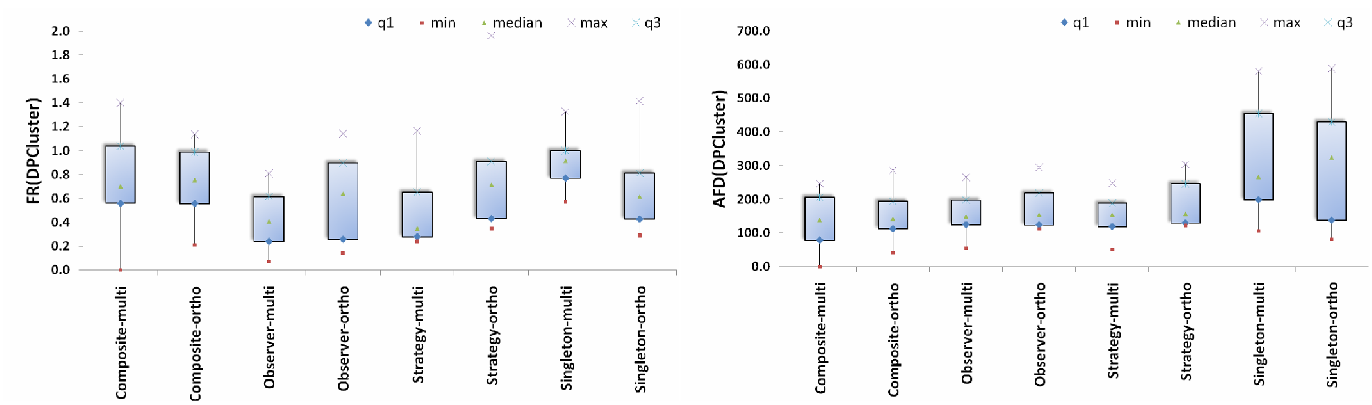
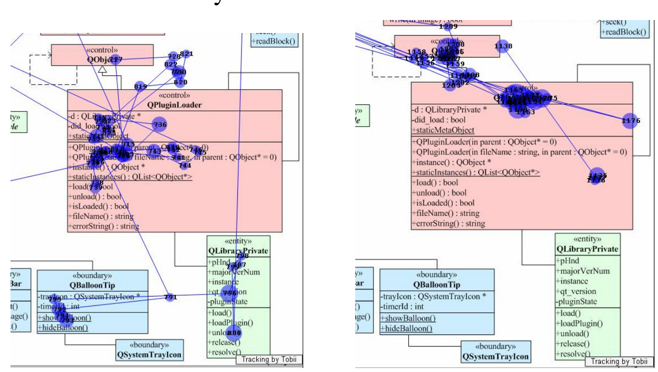


Figure 4. The fixation rate (left) and average fixation duration in ms (right) for the design pattern cluster (DPCluster) AOI.

D. *Qualitative Analysis*

A qualitative analysis is conducted using heat maps and gaze plots to answer the third research question (RQ3). A heat map is a technique to visualize gaze behavior of a group of subjects. A heat map is superimposed on top of the stimulus and highlights areas where subjects have been looking. Red indicates the highest percentage and green indicates the lowest. A gaze plot displays a static view of the eye gaze data for each diagram. It is useful to visualize scan paths. A scan path is a directed sequence of fixations. A fixation is illustrated using a circle where the radius represents the length of the fixation. These maps are best viewed in color.

One example of where experts differ from novices is in the identification of the Singleton pattern. See Figure 5. for an example of a gaze plot. It shows an expert looking at the attributes and methods to determine the answer. The novice on the other hand mainly focuses on the class name.



a) expert b) novice

Figure 5. Portion of a stimulus comparing an expert and novice gaze plot for Singleton pattern.

a) multi-cluster layout (JHotDraw) b) orthogonal layout (JUnit)

Figure 6. A heat map based on fixation length of all fifteen subjects for the Observer pattern

A cumulative heat map of all fifteen subjects is shown in Figure 6. for the Observer pattern in the layouts compared. The multi-cluster layout on the left is from *JHotDraw* and the paired diagram on the right in orthogonal layout is from *JUnit*. The highlighted areas correspond to the relative fixation length. These maps clearly show the difference in time spent by all subjects in the two different layouts. Eye fixations are mainly seen on the classes participating in the design pattern for the multi-cluster layout, whereas in the orthogonal layout, almost half the number of classes in the diagram are looked at for a considerably more time compared to the multi-cluster layout. This concurs with the AFD(nonDP) measure for the Observer pattern (*p-value*=0.001), where subjects spend more time in phase one of the task. From Figure 6. we can see the visual effort in fixation length is clearly higher for the orthogonal layout. This indicates that classes participating in a design pattern in the multi-cluster layout act like visual beacons drawing the attention of subjects thereby allowing them to complete the task in less time.

E. *Similarities and Differences*

The main difference between this study and [18] is in the method of data collection. Here we use an eye tracker to gather data and compare and contrast the findings of the questionnaire-based study. The eye-tracking allows a fine-grained analysis of the results. The subject systems and design patterns used are same as the previous study discussed in [18]. The main difference is in the layouts presented to the subject in each pattern. In this study, the layout was swapped for each pattern. This was done in order to test the other untested half of the previous study. The only difference is in the fourth column, *Layout* in Table III. In this study, a different sample population was used. There were more knowledgeable experts with respect to design patterns.

F. *Discussion*

The multi-cluster layout has a positive effect on accuracy, speed and visual effort needed to solve the role detection tasks. In particular, the Strategy and Observer patterns benefitted the most. We could not reject the null hypotheses for the Composite and Singleton patterns.

The Strategy pattern consists of an aggregation and generalization relationship. The aggregation is not within the hierarchy like the Composite pattern. Since the aggregation is not within the same hierarchy in the Strategy pattern, the class participating in the *Context* role tends to be placed further away in the orthogonal layout. The multi-cluster layout on the other hand, recognizes cohesive clusters and places classes with aggregations closer together even though this might violate an aesthetic criteria of placing all children at the same level under the parent. This is the main factor that causes the Strategy pattern to have a higher accuracy with less time spent in role detection for the multi-cluster layout. Since the Composite pattern has it's aggregation within the hierarchy, it tends to be positioned similarly in both layouts. The same reasoning applies to the Observer pattern, where the orthogonal layout tends to place the Subject and Observer connected by an association further apart, requiring more effort in tracing.

Considering all the visual effort measures, we find more support for the Strategy and Observer patterns using the multi-cluster layout. Results also indicate more time spent looking for classes (phase one: AFD(nonDP)) involved in the design pattern in the orthogonal layout. Based on the eye gaze data, we find that novices and experts had different techniques to identify patterns and roles. Novices used a template matching method and tried to match the template to the diagrams occasionally looking at attributes and methods. Experts focused on method names and attributes in addition to the class names. This is more evident in the Singleton and Observer patterns. In order to identify Singletons, two criteria need to be met a) a self-association, and b) static instance. The novices only looked for self associations and this by itself does not indicate a Singleton class. Experts looked for the static instance first, which is the main reason behind the self association. It is important to use both fixation counts and gaze duration to determine the effort since, based on our results, we find that even though the fixation counts may not be significantly different, the duration might be and vice versa.

The post questionnaire collected information about the difficulty level of the systems used and the role detection in design patterns. None of the subjects (except one) were familiar with the design of the systems. None were aware of design pattern usage in the systems. The Singleton pattern was considered to be the easiest, followed by the Composite pattern. The Strategy and Observer patterns were ranked at a higher level of difficulty. All three systems were rated at the same average difficulty level indicating that the results are not attributed to system but rather on the layout itself.

V. THREATS TO VALIDITY

This section discusses threats to validity and measures taken to minimize them.

1) Internal Validity

To minimize learning effects, a different subject system was used for each layout due to the within-subjects nature of this study. The study was designed to be completed in less than 20 minutes to avoid fatigue effects. Each role detection question did not contain any other patterns i.e., Strategy detection diagrams did not contain a Composite pattern to avoid any confounding factors. Since this is an eye-tracking study, the reading method used by subjects might affect the results of detecting the roles of design patterns. A top-down scan versus a left-right scan of the diagram might affect the results. However, we did notice that subjects looked at almost all classes in the diagram before making their choice. The manually engineered diagrams might pose a threat, hence the number of crossings were kept the same across layouts. The number of classes and relationships were also the same across layouts. Another threat is the possible overlap in areas of interest. Sometimes, more than one object was part of the area of interest, making it difficult to know which object was actually being looked at. This was due to the very nature of the UML class diagram layout. Care was taken to minimize overlaps where possible. One way to recognize patterns is by certain naming conventions. Both layouts containing the patterns had similar naming conventions, making them equally easy or difficult.

2) External Validity

We used students and faculty as our sample population. They were all familiar with design patterns at varied levels of expertise. This allowed us to compare novices with experts. Many of the subjects worked in industry and are comparable to senior developers. Another concern with regards to external validity is with respect to representative tasks. The tasks were based on real open-source systems (not toy applications) and hence more representative of design pattern usage in software.

3) Construct Validity

Since visual attention is related to mental processing of the information [11], the measures derived from the fixation counts and durations should be valid. Eight measures were used to avoid mono-method bias. The visual effort measures for duration, use an average since the areas of interest are fairly large compared to eye tracking studies done in psychology where the area of interest is a word. Another option would be to use the sum to increase the power of the statistical tests.

4) Conclusion Validity

Due to low sample size, we use the paired Wilcoxon test for hypotheses testing. ANOVA was not used for determining interaction effects with Experience, due to low sample size and non-normality of some of the data.

VI. RELATED WORK

This section reports on empirical work done in assessing UML class diagrams via eye tracking and questionnaires. Guehéneuc [8] used a head-mounted eye tracking system, EyeLink II, to investigate the comprehension of UML class diagrams. Areas of interest include class bounding boxes that measure fixation and saccade aggregation. The results indicate that initially, software engineers browse through the class diagram in a random fashion to identify most useful parts. They then focus on parts related to the question asked. They found that software engineers do not use relationships such as inheritance, dependency, association, aggregation and composition which was a surprisingly result. This might be due to the nature of the questions and the simplicity of the class diagrams used.

Yusuf et al. [22] conducted an eye-tracking study to assess the comprehension of class diagrams using the Tobii eye tracker. Their study was an extension of our pilot study [1]. The use of layout, color, and class stereotypes (control, boundary and entity) were assessed to determine their effectiveness in UML and design related tasks. The results indicate variation of eye movements between experts and novices in both UML and software design ability. Class stereotypes also played a role in the answering of the questions. They found the orthogonal layout needed more effort in terms of the average number of fixations, with the multi-cluster layout requiring the least. This study was replicated with comparable results in [17] using an online timed questionnaire.

Jeanmart et al. [10] conducted a study on the effect of the Visitor design pattern on comprehension using an eye tracker. The study considers two types of tasks: comprehension and modification. Three design alternatives were used: diagrams with no patterns, diagrams with the canonical layout and diagrams with the pattern in a modified layout. One of the results was that the inclusion of the Visitor pattern in a class diagram plays a role in maintenance tasks. No significant difference was found for the comprehension tasks. In particular, the Visitor pattern layout in canonical form as presented in Gamma et al. [7] required less effort from developers. This is the first study besides our work that investigates layout in UML class diagrams. The results from their work and [18] support the results of the above study for role detection tasks of design patterns.

Eichelberger et al. [5] presents visual guidelines based on their previous work [4] for the aesthetic quality of UML class diagrams as a framework to improve the quality of UML class diagrams. To validate this, they describe a pilot study to determine the effect of the guideline rules on comprehension. This is the first recent pilot user study besides the controlled experiments conducted by the authors of this paper, that attempts to validate UML class diagram layouts. A set of diagrams consisting of 13 class-like elements and 12 relationships were drawn. The base diagram followed all the visual rules, whereas the five modified diagrams violated exactly one of five rules. Results indicate that the analyzed layout rules have a small effect compared with other characteristics of various diagrams used. There are three main things that differentiate our work from this study. First, they do not consider the semantic closeness of classes and focus mainly on UML notation variants. Second, the tasks presented in this study and our prior studies [1, 16-18] are more fine-grained software maintenance tasks whereas the tasks in their study are related to UML notation such as changing the relationships between existing classes or naming the classes derived from a parent class. Third, the diagrams in our study are representative of real systems since they are based on reverse-engineered designs. The diagrams used in [5] were not based on real systems and included very few attributes and method names. We consider their approach to be simplistic, albeit valid based on the goal of their experiment.

We conducted a controlled experiment empirically validating the orthogonal and multi-cluster layouts in six software task categories [16]. The multi-cluster layout achieves higher accuracy and takes less time than the orthogonal layout for a majority of the task categories, especially difficult and challenging ones.

With respect to stereotypes, Kuzniarz et al. [12] conducted a study investigating the effect of stereotypes in UML class and collaboration diagrams within the telecommunication domain.

The results of this study statistically prove that the use of stereotypes helped in system comprehension. Ricca et al. [15] conduct a series of four experiments that analyze the effect of using Conallen's stereotypes for web application comprehension. The stereotyped diagrams reduce the gap between high and low skilled subjects.

Purchase et al. [14] conducted empirical studies to determine important aesthetic criteria such as minimizing bends for class diagrams. The results were inconclusive but the authors point out the need for semantic grouping of elements in diagrams. Sun et al. [19] propose graph layout criteria based on perceptual segregation. They found symmetry, orientation, and contours to be important factors in recognition. Eiglsperger et al. [6] and Gutwenger et al. [9] present automated layout algorithms for class diagrams based on graph aesthetic criteria such as minimizing edge crossings and bends. Our work focuses on UML class diagram layouts based on architectural importance. It takes class importance into consideration in layout positioning. We use real open-source systems and realistic tasks to validate the layout schemes.

VII. CONCLUSIONS AND FUTURE WORK

An eye-tracking study is conducted to determine the effect layout has on the detection of roles in design patterns. This was a replication of the questionnaire-based study conducted earlier by swapping the type of layout for each pattern. Visual effort is determined via a set of eight measures and provides an objective metric to measure the quality of UML class diagram layouts. Both studies report higher accuracy for role detection in the multi-cluster layout in the case of the Strategy pattern. In addition, this study also reports a higher accuracy for the Observer role with the Observer pattern approaching significance. All four patterns report lower time spent on task in the multi-cluster layout. Results also indicate a lower visual effort in design pattern clusters for the Strategy and Observer patterns in the multi-cluster layout. In future work, we plan to expand the study to include other design patterns and the relationship of class stereotypes in design pattern role detection with respect to maintenance tasks. Investigating the level of detail with respect to layout is another area of future work. In this study, we did not analyze saccadic movement. One possibility is that frequent saccades may imply quick incorrect conclusions. This is also left as a future exercise.

ACKNOWLEDGMENT

We would like to thank Dr. David Robbins for assisting in the use of the Tobii eye tracker as well as every single individual who participated in this study.

REFERENCES

[1] Andriyevska, O., Dragan, N., Simoes, B., and Maletic, J. I., "Evaluating UML Class Diagram Layout based on Architectural Importance", in VISSOFT, Hungary, Sept 25, 2005, pp. 14-19.

[2] Booch, G., Jacobson, I., and Rumbaugh, J., The Unified Software Development Process, Addison-Wesley, 1999.

[3] Brooks, R., "Towards a Theory of the Comprehension of Computer Programs", International Journal of Man-Machine Studies, vol. 18, no. 6, 1983, pp. 543-554.

[4] Eichelberger, H., "Nice Class Diagrams Admit Good Design?" in ACM Symposium on Software Visualization (SoftVis), San Diego, USA, Jun 11-13 2003, pp. 159-167.

[5] Eichelberger, H. and Schmid, K., "Guidelines on the Aesthetic Quality of UML Class Diagrams", Information and Software Technology, vol. 51, no. 12, 2009, pp. 1686-1698.

[6] Eiglsperger, M., Kaufmann, M., and Siebenhaller, M., "A Topology-Shape-Metrics Approach for the Automatic Layout of UML Class Diagrams", in SoftVis, 2003, pp.189-198.

[7] Gamma, E., Helm, R., Johnson, R., and Vlissides, J., Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1995.

[8] Guéhéneuc, Y.-G., "TAUPE: towards understanding program comprehension", in 16th IBM Centers for Advanced Studies on Collaborative research, Canada, Oct 06, pp. 1-13.

[9] Gutwenger, C., Jünger, M., Klein, K., Kupke, J., Leipert, S., and Mutzel, P., "A New Approach for Visualizing UML Class Diagrams", in SoftVis, 2003, pp. 179-188.

[10] Jeanmart, S., Guéhéneuc, Y.-G., Sahraoui, H., and Habra, N., "Impact of the Visitor Pattern on Program Comprehension and Maintenance", in 3rd International Symposium on Empirical Software Engineering and Measurement, Lake Buena Vista, Florida, Oct 15-16 2009, pp. 69-78.

[11] Just, M. and Carpenter, P., "A Theory of Reading: From Eye Fixations to Comprehension", Psychological Review, vol. 87, 1980, pp. 329-354.

[12] Kuzniarz, L., Staron, M., and Wohlin, C., "An Empirical Study on Using Stereotypes to Improve Understanding of UML Models", in 12th Intl Workshop on Program Comprehension (IWPC) 2004, pp. 14-23.

[13] Purchase, C. H., Allder, J.-A., and Carrington, D., "Graph Layout Aesthetics in UML Diagrams: User Preferences", Journal of Graph Algorithms and Appl, vol. 6, no. 3, 2002, pp. 255-279.

[14] Purchase, C. H., McGill, M., Colpoys, L., and Carrington, D., "Graph Drawing Aesthetics and the Comprehension of UML Class Diagrams: An Empirical Study", in Australian Symp. on Information Visualisation, Sydney, 2001, pp.129-137.

[15] Ricca, F., Di Penta, M., Torchiano, M., Tonella, P., and Ceccato, M., "How Developers' Experience and Ability Influence Web Application Comprehension Tasks Supported by UML Stereotypes: A Series of Four Experiments", IEEE Trans. on Software Engineering, vol. 36, no. 1, Jan/Feb 2010, pp. 96-118.

[16] Sharif, B. and Maletic, J. I., "The Effect of Layout on the Comprehension of UML Class Diagrams: A Controlled Experiment", in VISSOFT, Canada, Sept 25 2009, pp. 11-18.

[17] Sharif, B. and Maletic, J. I., "An Empirical Study on the Comprehension of Stereotyped UML Class Diagram Layouts", in 17th IEEE Intl Conf on Program Comprehension (ICPC), Vancouver, BC, Canada, May 17-19 2009, pp. 268-272.

[18] Sharif, B. and Maletic, J. I., "The Effects of Layout on Detecting the Role of Design Patterns", in 23rd IEEE-CS International Conference on Software Engineering Education and Training (CSEE&T), Pittsburgh, Mar 9-12 2010, pp. 41-48.

[19] Sun, D. and Wong, K., "On Evaluating the Layout of UML Class Diagrams for Program Comprehension", in 13th IEEE Intl Workshop on Prog. Comp., Missouri, USA, 2005, pp. 317-328.

[20] Sutton, A. and Maletic, J. I., "Recovering UML Class Models from C++: A Detailed Explanation", Information and Software Technology, vol. 49, no. 3, Jan 2007, pp. 212-229.

[21] Wohlin, C., Runeson, P., Host, M., Ohlsson, M. C., Regnell, B., and Wesslen, A., Experimentation in Software Engineering - An Introduction., Kluwer Academic Publishers, 2000.

[22] Yusuf, S., Kagdi, H., and Maletic, J. I., "Assessing the Comprehension of UML Class Diagrams via Eye Tracking", in 15th Intl Conf on Program Comprehension, Canada, June 26-29 2007, pp. 113-122.