Exploratory visualization techniques may complement statistical data analysis, helping users to understand and treat data from empirical studies. Visualization becomes particularly interesting as data sets grow large and more diverse. This paper discusses how visual representations and exploratory data visualization may be applied to support and enhance analysis of data sets produced in experimental Software Engineering. Information Visualization techniques are used to analyse data from an experiment comparing different reading techniques, allowing us to identify advantages and limitations of visual approaches as compared with traditional statistical techniques.

1. Introduction

In order to successfully attain project goals and deadlines software development teams must choose suitable models and supporting techniques [5]. Empirical Software Engineering attempts to evaluate models and techniques, registering how they perform in practical contexts, with the goal of establishing a knowledge database to support decision making for development. As such, empirical studies have been conducted aimed at providing evidence of the quality and productivity of software development methods, techniques and tools [2, 6, 9, 10].

Experimentation processes are typically conducted to validate previously formulated hypotheses. Hypotheses state an assumption on how dependent variables are influenced by the independent ones. Thus, in an experimental design the independent variables are isolated to investigate if the hypotheses hold, usually determined by a statistical analysis. As data is gradually accumulated, it may be difficult to observe non-anticipated relationships and patterns in the data applying only standard statistical techniques. Visual representations may help data analysts to convey information better, and visualization techniques provide an alternative approach to explore data sets produced in empirical software engineering.

In this work visualization techniques have been employed to support knowledge presentation, hypothesis-oriented data analysis (confirmatory visualization) and undirected data exploration (exploratory visualization). Data collected in one experiment replication conducted in the scope of the Readers Project [11], that congregates Brazilian and American researchers on an effort to produce and integrate a large body of results from controlled experiments on families of technologies, is used to illustrate such uses.

This paper is organized as follows: in Section 2 we discuss work on Empirical Software Engineering and describe the experiment used to illustrate the potential role of visualization in data analysis; in Section 3 we present a brief overview of Information Visualization; in Section 4 we describe how Visual Data Analysis has been applied both in hypothesis-driven and exploration-driven data analysis. In Section 5 we provide a discussion and perspectives for further work.
of a family of reading techniques for detecting defects in software requirements documents. PBR provides a process to review requirements documents in which the reviewer assumes one particular perspective: Designer (D), Tester (T), or User (U). Depending on the perspective, the reader must follow specific guidelines to conduct the revision process. Experiments have thus been designed to compare PBR against other reading techniques, particularly with the Checklist approach for defect detection. Several experiments have been replicated in the scope of the Readers project, producing data on PBR [11].

The original experimental design of the PBR experiment, run at the University of Maryland, addressed questions such as: 1) Do teams applying PBR detect more defects than teams applying Checklist? 2) Do individual reviewers using PBR and Checklist find more defects? 3) Does the reviewer’s experience affect his or her effectiveness? Shull et al. [11] pointed out that the original experimental design left some open questions, and the experimental design for a set of replications of the original experiment extended the roll of questions to be investigated, such as: 4) Does a reviewer individually find out different defects applying PBR and Checklist techniques? 5) Do the PBR perspectives have the same effectiveness and efficiency? 6) Do the PBR perspectives find different defects?

Following the design specified in the Lab Package, shown in Figure 1, subjects were divided into two groups for the experiment. In the first day, subjects in both groups used the Checklist technique to review one of two requirements specification documents: Automated Teller Machine (ATM) or Parking Garage (PG). In the second day, each subject was trained in one of the PBR perspectives – Designer, Tester or User – to review the document that she/he had not revised yet: those who had reviewed the ATM document applied one PBR perspective to review the PG requirements document, and vice-versa. For each defect occurrence observed the subject should register the page on the correspondent requirement document and classify the defect according to a given taxonomy: Ambiguous Information (A) – Information is ambiguous; Inconsistent Information (II) – Two sentences contradict each other; Incorrect fact (IF) – Some sentences assert a fact that cannot be true; Extraneous Information (E) – Information is provided, but is not needed or used; Miscellaneous Defect (MD) – Other defects; Omission (O) – Necessary information has been omitted.

3. Information Visualization

Visual representations allow data analysts to use their visual capability to recognize patterns and structures embedded in data, and are also useful to convey known information about processed data. Visualization techniques can be categorized, according to the type of task supported, in presentation, confirmatory analysis and exploratory analysis. Presentation techniques assume that facts to be presented are known and fixed a priori. Confirmatory visualization techniques are useful when the analyst has some hypotheses about the data, and the goal of the visualization is to support confirmation or rejection of these hypotheses – the visualization may motivate (or not) additional statistical analyses on the data. Exploratory techniques create representations from raw data, and are useful when there are no a priori hypotheses – for example, the data set is new to the user, who has little idea of what to search for in the data, or may be interested on pre-processing to improve data quality for further input into analytical algorithms and tools. With interactive exploration, a user conducts an undirected search for structures, and gradually forms mental hypotheses that may be confirmed by appropriate visualizations or statistical analysis [1]. These different visualization categories may be applied to data from empirical studies to support different analysis tasks.

Figure 1. Experimental Design [2]

<table>
<thead>
<tr>
<th>First Day</th>
<th>Group A</th>
<th>Group B</th>
<th>Checklist</th>
</tr>
</thead>
<tbody>
<tr>
<td>Day</td>
<td>Training</td>
<td>Training</td>
<td>ATM (D)</td>
</tr>
<tr>
<td></td>
<td>ATM (T)</td>
<td>PG (U)</td>
<td>(U)</td>
</tr>
<tr>
<td></td>
<td>Theory PBR</td>
<td>Training</td>
<td>PBR</td>
</tr>
<tr>
<td></td>
<td>Training</td>
<td>PG (U)</td>
<td>(U)</td>
</tr>
</tbody>
</table>

Figure 2. Visual Analysis Process [3]

A typical Visual Analysis Process follows the pipeline described by Card et al. [3], depicted in Figure 2. The pipeline describes the tasks and interactions for producing a human perceivable, interactive visual representation from raw data. In this process each arrow indicates multiple transformations over the data: once raw data is “cleaned” and instances and relationships organized into a data table, it is possible to create Visual Structures. These Visual Structures – combining spatial substrates, marks and graphical properties – may be interactively modified by the user in order to reach his/her goals. Views from the visual structures are obtained by setting graphical parameters, such as scaling and clipping. One may identify two phases in the process: Data Treatment and Visual Treatment. Although
connected, these phases occur in different spaces (the data space versus the representation space) and, therefore, require different types of support. We observed that the ability to interact with visual representations was a major resource to explore possibilities and draw conclusions.

4. Visualization Applied to Empirical Software Engineering: PBR Data

Visualization techniques may assist in different stages of a data analysis process with different roles. One may identify the major stages of a process supported by visual analysis tools starting from Wohlin et al.’s [12] description of a statistical analysis process: a Pre-Analysis stage involves data processing and treatment, including data set reduction when applicable; Hypothesis Verification, which is in the core of the analysis; and two additional stages that may rely on heavy support from visual techniques, Synthesis – which involves presenting results – and Meta-Analysis – which refers to analyzing data from multiple experiments or replications.

Figure 3 shows our view of the potential application space of visualization techniques in the analysis of empirical data, considering three axes representing the type of user (Designer of the experiment, Replicator or Subject), the stage in the analysis process (Pre-processing, Hypothesis Verification, Synthesis or Meta-Analysis), and the nature of the tasks to be supported (Presentation, Confirmation, Exploration). The front plane, for example, shows the potential usefulness of visual techniques to an experiment Designer. Exploratory visualization techniques are potentially useful in Pre-processing, Hypothesis Verification and Meta-analysis, and Visual Presentation techniques are useful in the Synthesis stage. Visual Presentations may also convey information to Subjects and Replicators, as well as to the Designers themselves. The size of the circles drawn in the figure is proportional to potential usefulness. Thus, the previous examples illustrate that techniques are likely to be very useful to Designers in the corresponding stages. Similarly, the middle plane shows that an experiment Replicator is likely to benefit a lot from Presentation techniques in the Synthesis stage and from Confirmatory visualization in the Analysis stage, though a replicator is usually not concerned with Meta-Analysis. The back plane shows that Subjects might resort to visual techniques, for example, to understand results and evaluate their own performance.

Considering the above scenario, visual analyses conducted in this study concentrate mainly on the front and middle plane regions, considering the experiment Designers’ and Replicators’ perspectives. Analyses were conducted with two different goals. First, confirmatory visualizations that could actually support qualitative “hypothesis” confirmation or rejection were produced. Here, interactive visualizations were the main resource to identify and group the attributes of interest to visually “show” how the data behaves regarding a certain hypothesis. Once the relevant information is identified, it may be conveyed by a proper visual presentation organized to show the target (dependent) attribute in relation to the secondary (independent) ones. At this point, simple presentations, such as pie and bar charts, usually suffice. Secondly, exploratory analyses were conducted trying to look at the data without directing the search by hypotheses. Undirected exploration was limited, however, as the study was restricted to a small data set from a single replication.

4.1. Hypothesis-Driven Visualization

Initial visualizations were inspired on the Questions posed, trying to visually identify as trends in a graphical representation the known statistical results. Focusing on Question 2 (Do individual reviewers using PBR and Checklist find more defects?), an example of a hypothesis-driven visualization that considers the type of the defects is presented in Figure 4. In this visualization, the horizontal axis ($x$) depicts defect type, the vertical axis ($z$) represents the defect identification numbers, and the reading technique is on the $y$ axis. This visualization allows observing the distribution of reported defects and their types, comparing the behavior of both PBR and Checklist techniques. One may observe, for example, that defects of the type Omission (O) have been found more frequently than other types and that very few defects of type Extraneous Information (E) have been found.

Focusing on Question 3 (Does the reviewer’s experience affect his or her effectiveness?), the visualization in Figure 5 allows observing the role of reviewers’ previous experience and background on performance. Subjects’ experience as software Developer, Manager, Tester and An-
Figure 4. Defects found by Technique and Defect Type

Figure 5. Defects detected by Document and Technique – color maps average Experience

Figure 6. Defects detected by subject (PG)

Figure 7. Defect distribution in PG document

4.2. Exploratory Data Visualization

Although limited within the scope of data from a single experiment, we investigated how visualization could assist an undirected exploratory process on the data. In this context, we tried to produce visualizations capable of conveying potentially interesting and unknown relationships amongst independent experimental variables. Figure 8 shows a Parallel Coordinates [7, 8] visualization of the PBR replication data, focusing on the attributes that register subject’s previous experience as Manager, Developer, Analyst, Tester, Using Requirements and Writing Requirements. The range of values for these four attributes was normalized and experience values are expressed in months.

Figure 8 shows that 14 out of 18 subjects have exactly the same experience on Using Requirements and Writing...
Requirements, and four of the subjects have quite similar experience in general. It might be interesting to verify if this holds in other replications, as this information might be considered for attribute reduction in future replications. Carver [4] has discussed some specific skills that may affect software development experience, suggesting a study about their particular influence on subject’s effectiveness. A visual analysis may help to decide whether it is worth to conduct further statistical analysis on all the experience metrics collected from subjects. For example, it may not be worth to consider separately the experiences on Using Requirement and Writing Requirement.

![Figure 8. Experience of Subjects](image)

5. Conclusions and Further Work

This work is a first initiative towards introducing visual techniques in the analysis of data from empirical studies in software engineering. We observed that the application of such techniques is not immediate, requiring training on techniques and domain knowledge to avoid creating naïve presentations, that may induce mistaken observations and conclusions. Although visual techniques do not substitute statistical analysis, our case study illustrates that visualizations may improve interpretation of the experimental data. Exploratory visualization can assist not only in hypothesis verification, but also in observing trends in data. In both situations it may avoid repeated statistical analysis that may not be worth running, as some facts and trends can be observed with visual interactive techniques and multiple graphical representations. Visual representations bring agility and flexibility to data analysis. Also, some possibilities for enhancing the Lab Package were identified.

The major effort to conduct a meta-analysis is quite likely to benefit from visual analysis. As further work, we are investigating how visual data analysis can support meta-analysis and deal with the difficulties introduced by different data organizations. Also, new hypotheses might be proposed for future experiments using exploratory visualization and visual data analysis.

Acknowledgments

The authors acknowledge the financial support of FAPESP – the State of São Paulo Research Funding Agency, Grant 01/13993-0 and CNPq, Grant 521931/97-5 and Grant 680156/01-1.

References