VISUALIZING MUTATIONS OF A VIRUS SEQUENCE

by

Mahshid Zeinaly Baraghoush

B. Sc., Sharif University of Technology, 2009

a Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in the School of Interactive Arts and Technology Faculty of Communication, Art and Technology

© Mahshid Zeinaly Baraghoush 2012 SIMON FRASER UNIVERSITY Fall 2012

All rights reserved.
However, in accordance with the Copyright Act of Canada, this work may be reproduced without authorization under the conditions for “Fair Dealing.” Therefore, limited reproduction of this work for the purposes of private study, research, criticism, review and news reporting is likely to be in accordance with the law, particularly if cited appropriately.
APPROVAL

Name: Mahshid Zeinaly Baraghoush
Degree: Master of Science
Title of Thesis: Visualizing Mutations of a Virus Sequence

Examinig Committee:
Chair: Dr. Philippe Pasquier
       Assistant Professor

Dr. Chris Shaw
Senior Supervisor
Associate Professor

Dr. Lyn Bartram
Supervisor
Associate Professor

Dr. Halil Erhan
External Examiner
Assistant Professor

9 Nov 2012
Date Defended/Approved:
Partial Copyright Licence

The author, whose copyright is declared on the title page of this work, has granted to Simon Fraser University the right to lend this thesis, project or extended essay to users of the Simon Fraser University Library, and to make partial or single copies only for such users or in response to a request from the library of any other university, or other educational institution, on its own behalf or for one of its users.

The author has further granted permission to Simon Fraser University to keep or make a digital copy for use in its circulating collection (currently available to the public at the “Institutional Repository” link of the SFU Library website (www.lib.sfu.ca) at http://summit.sfu.ca and, without changing the content, to translate the thesis/project or extended essays, if technically possible, to any medium or format for the purpose of preservation of the digital work.

The author has further agreed that permission for multiple copying of this work for scholarly purposes may be granted by either the author or the Dean of Graduate Studies.

It is understood that copying or publication of this work for financial gain shall not be allowed without the author’s written permission.

Permission for public performance, or limited permission for private scholarly use, of any multimedia materials forming part of this work, may have been granted by the author. This information may be found on the separately catalogued multimedia material and in the signed Partial Copyright Licence.

While licensing SFU to permit the above uses, the author retains copyright in the thesis, project or extended essays, including the right to change the work for subsequent purposes, including editing and publishing the work in whole or in part, and licensing other parties, as the author may desire.

The original Partial Copyright Licence attesting to these terms, and signed by this author, may be found in the original bound copy of this work, retained in the Simon Fraser University Archive.

Simon Fraser University Library
Burnaby, British Columbia, Canada

revised Fall 2011
Ethics Statement

The author, whose name appears on the title page of this work, has obtained, for the research described in this work, either:

a. human research ethics approval from the Simon Fraser University Office of Research Ethics,

or

b. advance approval of the animal care protocol from the University Animal Care Committee of Simon Fraser University;

or has conducted the research

c. as a co-investigator, collaborator or research assistant in a research project approved in advance,

or

d. as a member of a course approved in advance for minimal risk human research, by the Office of Research Ethics.

A copy of the approval letter has been filed at the Theses Office of the University Library at the time of submission of this thesis or project.

The original application for approval and letter of approval are filed with the relevant offices. Inquiries may be directed to those authorities.

Simon Fraser University Library
Burnaby, British Columbia, Canada

update Spring 2010
Abstract

This thesis addresses a synthetic health-dataset, introduced at the IEEE VAST Challenge 2010. My research team participated in this contest to evaluate the pre-designed tool, "IMAS" using a benchmark dataset.

Learning from this contest, I designed an Information Visualization (InfoVis) prototype, "FilooT" to gain a better understanding of that dataset. Following the Nested Model for Visualization Design, my thesis’ qualitative methodology consists of a design study and evaluation.

To make an effective design, I followed well-cited InfoVis principles of perception and cognition. I also utilized prior knowledge produced by the proposed solutions that had been tackled the contest’s dataset.

To understand the tool’s design capabilities for target domain analysts, I observed domain-users’ reactions to FilooT in a User-Experience scenario. The findings of the study indicated how analysts employ each of the visualization and interaction designs in their Bioinformatics’ task-analysis process. The critical analysis of the results inspired design informing suggestions.

Keywords Human Computer Interaction; Information Visualization; Visual Analytics; Visual Encoding and Interaction Design; User Experience; IEEE VAST Challenge
To my beloved parents,

and

To all the people who valued the luxury of learning in my life.
“The ending of all the good stories suggests a new beginning”

— Dr. Carolyn Mamchur, Professor, Language Arts
Acknowledgments

First, I express my deepest gratitude to Dr. Christopher Shaw, my senior supervisor at SFU, for the initial inspiration of this work and for his continuous feedback and support throughout the different phases of this project.

In the same manner, I would like to thank Dr. Lyn Bartram, my supervisor, for her interesting Knowledge Visualization course at SFU, which helped me to express my first visualization ideas in a second language. She has been also a great inspiration for me as a female professor in Computer Science. Also, thanks to Dr. Halil Erhan for accepting to be the external examiner in my committee.

In addition, I thank Dr. Tamara Munzner for her insightful comments on the design of this project. I also thank Dr. Cydney Nielsen from BC Cancer Agency for her valuable comments on the initial prototype.

I sincerely appreciate my friends and colleagues at the SIAT visual analytics lab, Saba Alimadadai, Evan Dickinson, Mina Soltangheis, and Ji-Dong Yim, for being constantly available to encourage, to inspire, and to make my experience more joyful.

I thank my friends for life who helped keep me on track, especially through the past year: Pooya Amini, Mona Harati, Maryam Saberi, Yasaman Sefidgar, Hasti Seifi, Arman Tavakoli, Aylin Tavakoli, and Negin Zarnegar.

To Abtin Rasoulian: Many thanks for your enduring help, and for doing seven years of academic journey with me.

I simply could not travel through life to reach my ambitions without all the people who believed in me, my brother’s encouragements, and my parents’ unconditional love.
Contents

Approval ii
Abstract iii
Dedication iv
Quotation v
Acknowledgments vi
Contents vii
List of Tables xiii
List of Figures xiv
Preface xvii

1 Introduction 1
  1.1 Information Visualization Process 2
  1.2 Purpose 5
  1.3 Problem 6
    1.3.1 Tasks 7
  1.4 Approach 8
  1.5 Thesis Outline 9

2 Literature Review 10
  2.1 Technical Background 10
# 2.1 Biological Data Visualization

- Biological background ............................................ 10
- The data-set/task-set .............................................. 11

# 2.2 Positioning This Study in Related Work .............................. 16

- Multiple Sequence Alignment Viewer ................................ 16
  IMAS (Interactive Multigenomic Analysis System) .................. 18
  Jalview (A Multiple Alignment Editor written in Java) ............ 18
- Matrix Viewer .......................................................... 21
- Hierarchy representations ............................................ 21
- Network representations ............................................. 25
- Overview+Detail ....................................................... 25
- Focus+Context .......................................................... 26
- Other VAST Challenge Solutions .................................... 26
  ManyNets ................................................................ 27
  Cognizant BFS Innovations prototype ................................. 28
  GeneTracer .................................................................. 30
  SequenceView ................................................................ 31

# 3 Methodology: Design ......................................................... 33

- Technical Design Background .......................................... 33
  Interaction Techniques in Visualization .............................. 36

## 3.1 Information Visualization ............................................. 33

### 3.1.1 Interaction Techniques in Visualization ....................... 36

## 3.2 Applying Technical Design Background to FilooT Design ....... 37

### 3.2.1 Domain Data Translation ........................................... 38
  Translating the DNA Information Table .............................. 38
  Translating the Table of Disease Characteristics .................. 39

### 3.2.2 Domain Data Visual Encoding ................................... 39

### 3.2.3 Task Translation .................................................... 41
  Task 1 Translation ................................................................ 41
  Task 3 Translation ......................................................... 41

### 3.2.4 Visual Encodings and Interaction Design ................. 42
  Sorting ........................................................................ 42
  Filtering ...................................................................... 44
D.4.2 Sequence Diagrams ........................................ 130
D.4.3 Compose Structure Diagrams ............................. 130

Bibliography ........................................................................ 141

Index .................................................................................. 146
List of Tables

1.1 Sequence Characteristics Table. ........................................ 7

3.1 A sample part of the virus strain sequences table. ................. 38
3.2 A sample part of the sequence characteristics table. ............. 38
3.3 Translated sample part of the virus strain sequences table. .... 39
3.4 Translated sample part of the sequence characteristics table. ... 39
3.5 Complementary Relationship ............................................. 56

5.1 The organization of the users’ ideas .................................. 82
5.2 pre-questionnaire study results ......................................... 94

A.1 Sequence Characteristics Table ....................................... 99

B.1 Top mutations based on correlation to symptom severity .......... 102
B.2 Task 4 Official solution .................................................. 103

D.1 View Selected Subseq. Menu Options: UseCase Description .... 113
D.2 Create Feature: UseCase Description .................................. 113
List of Figures

2.1 An example of the Alignment Profile ........................................ 12
2.2 SFU-SIAT-IMAS team submission, IMAS snapshot solution for task 1 .... 14
2.3 ManyNets Network view ............................................................ 15
2.4 Different colour codes are used to encode A, C, T and G .................. 17
2.5 The shading approach in multiple sequence alignment viewers .......... 17
2.6 IMAS multiple alignment view ................................................... 19
2.7 Jalview features. ........................................................................ 20
2.8 Types of nodeLink tree Visualization. ........................................... 22
2.9 Types of Adjacency Diagrams ...................................................... 23
2.10 Types of Enclosure Diagrams ....................................................... 23
2.11 Sunburst diagram from Noblis team submission. ............................. 24
2.12 Two different graph layout ......................................................... 25
2.13 ManyNets tabular view .............................................................. 27
2.14 Solution given by team Cognizant BFS Innovations ....................... 29
2.15 Solution given by team Gene Tracer submission. ........................... 31

3.1 Mackinlay’s Ranking list of visual properties for different data type ...... 34
3.2 The visual encoding of the first table ............................................ 40
3.3 Vischeck runs to test colour blindness ........................................ 40
3.4 The initial design of Matrix View employs bar lengths. .................... 42
3.5 Design of the Matrix View. ......................................................... 43
3.6 How sorting interaction works .................................................... 44
3.7 How to make a new column from existing ones. ............................... 45
3.8 Hiding the columns. ................................................................. 46
3.9 Filtering the columns. ............................................................... 47
4.1 The Nested Layer framework for this study. Redrawn after Figure 1 of [27] . 77
Preface

My intended audience for this thesis documentation is someone who is knowledgeable about Information Visualization (InfoVis), Visual Analytics (VA) tools, and with some knowledge of Bioinformatics. Yet, this thesis aims to be accessible to those new to both Human Computer Interaction (HCI) and InfoVis fields especially at the graduate level. Therefore, I explained, "what visualization is" in chapter one and described the necessary background to understand the problem domain and the solution in chapter two.
Chapter 1

Introduction

Information Visualization (InfoVis) tools in different domains have been developed to enhance a better understanding of large complex data-sets. These tools employ human vision to solve the domain tasks rather than automatically producing a final solution that limits the discovery of insights in many aspects of the data [40]. In order to focus on the unique features of a data-set/task-set in a particular domain and open the possibilities of more general visualization design patterns emerging in that specific domain, many researchers design custom solution tools instead of utilizing a general framework.

Throughout this thesis, I address a highly acknowledged [4, 9, 12, 20, 22, 29, 45, 50] health and genetic data-set/task-set combination that is introduced as a benchmark in the Institute of Electrical and Electronics Engineers Conference on Visual Analytics Science and Technology (IEEE VAST) Challenge 2010. The VAST Challenge goal is to call Visual Analytics (VA) researchers to solve challenge problems, and to evaluate their tool using benchmark data-sets. In an attempt to gain a better understanding of that data-set, I built an InfoVis prototype called FilooT. FilooT is designed to be a part of the Interactive Multi-genomic Analysis System (IMAS) project [39], which is a general framework developed to enhance analysis of genomic data-sets. Following the Nested Model for visualization Design [27], my thesis’ qualitative methodology consists of a design study and evaluation.

As part of the design process, I justify each decision based on specific prescriptions in the InfoVis literature. Specifically, the Nested Process Model [27] framework guides my design study. Following that framework, I map data to a visual encoding by careful consideration of data-type, visual channel characteristics and known perceptual and cognitive principles [23, 51, 47, 40]. Another fundamental part of this tool is its interactivity with users, which helps
free up their cognition memory for higher level brain tasks [40]. Each user-interaction is designed based on a translation of the task from its original Bioinformatics domain into InfoVis vocabulary [27]. This tool utilizes other InfoVis concepts such as multiple views, overview-detail [7], aggregation methods, and focus-context [7], as well as animation for transitions [34].

In the IEEE VAST Challenge 2010, a number of researchers proposed solutions to tackle the data-set. Each solution solves an isolated part of the problem, but fails in a broader sense. To make a better combination of the solutions, I also utilizes the prior knowledge produced by the solutions that had been developed during the design phase of FilooT.

In order to choose an evaluation methodology for my design study [38], I define my goal to “understand how analysts employ each of the visualization and interaction designs in VAST Challenge analytic process”. As Ellis et al. [10] suggest, my research falls into the Formative category because the purpose of the study is to inform the design rather than summarizing the effectiveness of the tool (Summative Research [10]). Considering the formative nature of my research question, I follow a qualitative methodology for evaluating the designs. The tool’s design is refined many times in an iterative process based on InfoVis specialists’ feed-backs. Furthermore, in order to understand the design capabilities of the tool for Bioinformatics analyzers, the reactions of domain users to FilooT in a User Experience Scenario [21] is carefully observed.

1.1 Information Visualization Process

Analyzing a data-set refers to answering questions about the data, rejecting or accepting a hypothesis, forming a new hypothesis, and discovering new insights from the data. This understanding involves finding trends and patterns in the data, identifying outliers, and characterizing anomalies. Sometimes this analysis can be done manually, especially if the size of the data-set is relatively small. On the other hand, if the data-set is too large or too complex for individuals to manually process it, a computer system can help to generate answers to some of the specific questions. In this regard, other disciplines such as statistics or data-mining are used to help the analytic process [18]. However, data modelling may not be trivial for a computer due to problem complexity [40]. For example Multiple Sequence Alignment problems lead to NP-complete combinatorial optimization problems [46]. In those situations a domain specialist is required to look at the different aspects of the data.
In addition, by properly exploring the data, the target domain experts might discover some patterns or answers to some questions nonexistent prior to their discovery \[35\]. In these cases, Visual Analytics (VA) or Information Visualization (InfoVis) systems utilize human vision to give new understanding of the data.

A well-designed data representation is a requirement for proper communication with domain experts. For any single set of data, there are a number of different display options. The way that InfoVis designers visualize the data, depends on the kind of questions that the target domain users need to answer. As Ben Fry says,

“Great information visualization never starts from the standpoint of the data-set; it starts with the question.” \[13\]

The questions that designers ask in order to make a visualization tool could be either high-level or low-level \[1\]. The initial questions are more abstract and address high-level tasks performed by the domain users. For any given data-set, in order to characterize the domain questions, the researchers could interview or observe the target users working with the data-set \[27\]. In some cases there are benchmark tasks associated with data-sets that have already been validated with the target audiences. This is the case in my research, and I do not conduct a field study to determine high-level tasks.

After clarification of the high-level tasks, the next step is designing the visualization system. In this step, the researchers might realize the irrelevancy of an InfoVis solution and recommend other options such as automation of the workflow. If the designers choose to build a visualization system, they must follow a methodology to design visual encodings and interactions to address each of the high-level tasks.

In order to design such systems, one methodology that I also adopted in this research is that the researchers define the high-level task in InfoVis language by breaking down the original tasks into smaller low-level tasks that are defined in the InfoVis language and for which there are visualization techniques. Transforming the data-type and tasks from the target domain to data-types of InfoVis domain \[27\] aids the designers to free up their mind from the target domain concepts. For example, a real world problem which I address in this thesis is “finding mutations that increase the disease characteristics in different patients”. This problem can be transformed into “sorting” the patients according to their disease characteristics, “filtering” the mutations that are not important, and “highlighting” the mutations that seem to relate to each of the disease characteristics. “Sorting”, “filtering”
and “highlighting” comprise InfoVis’s generic low-level task vocabulary [1].

Sedlmair et al. [38] define the word Design as follows: “...design is the creative process of searching through a vast space of possibilities to select one of many possible good choices from the backdrop of the far larger set of bad choices. Successful design typically requires the explicit consideration of multiple alternatives and a thorough knowledge of the space of possibilities.” [38]. For example, there are many ways to address “sorting” data values. Choosing between alternative visualization techniques depends on many factors, such as how the users will apply the sorting results in the current step, as well as subsequent steps of their task completion process. In regards to selecting between alternatives, InfoVis has some guidelines and heuristic lists that are based on cognitive science and psychology. For example, I use the heuristic list that ranks visual channels according to different data-types (quantitative, nominal, and ordinal) [23] which enhances the choice of visual properties of marks in my design. Additionally, I use examples on perception and colour from Ware’s book [47].

It is worth mentioning that some of the existing visualization techniques for low-level tasks, require specific algorithms to work efficiently. For instance, the Bubble Sort algorithm can reduce increase the computational complexity and subsequently reduce the efficiency of a visualization tool (although it might not be so much an issue with faster machines). Given that, the InfoVis literature recommend separation of these algorithmic design from visualization design in order to distinguish between the evaluation methodologies [27]. As an example, for an algorithm design, one might choose to evaluate the design by analyzing the computational complexity of the algorithm [27]. In contrast, for visualization design the researchers could analyze the user’s comments on the system. This research study relies on available algorithms and it only focusses on visualization design details.

Following the InfoVis guidelines can make the design more effective. However, it is not recommended to follow all of them [6]. Although these guidelines should be kept in mind to avoid mistakes for best practice [6], they cannot guarantee that the design is effective for the target users. At this point it is important to validate our design using the domain analyzers’ expert advice, while following appropriate evaluation methodology.

The key to choosing the evaluation methodology is to understand the goal of the evaluation [8]. In general, there are two types of goals. One such goal is adapting the design for target users in an iterative process using expert advice. Another goal is assessing the design’s capabilities through the measurement of reducing tasks times and/or increasing
Accuracy [10]. The design evaluation of my study is directed by suggested methodologies from Lam et al. [21]. For my purpose, I selected the User Experience Scenario methodology in order to validate and inform the design with domain experts. Many evaluation methodologies require working prototypes in order to test design features. The possibilities for a working prototype include image prototypes and software prototypes. In this research, I turned most of the design ideas into a software prototype, and some ideas remained paper prototypes.

The qualitative and formative analysis of the results is used to critique the design and suggest improving directions. The findings of the user study (Chapter 5) are useful resources for researchers who may wish to incorporate these design features into their own tools. Additionally, the analysis of the user study may inform other researchers about the capability of their own designs (which may be similar to FilooT) and their limitations by comparing the designs to the one examined in this study.

1.2 Purpose

The purpose of this thesis is to demonstrate the process I have been engaged in to build and examine a tool called FilooT for helping Bioinformatics analyzers make sense of a biological data-set. FilooT is a digital prototype of my designs for exploring a public data-set/task-set.

In order to tackle the data-set, I adapt an already existing tool called IMAS [39] and expand one of its views to help analyzers understand the data-set/task-set. The replacement of FilooT with IMAS multi-alignment view is furthering this research and functions as a speculative project for future work.

The data-set/task-set are synthesized Bioinformatics health data, and there are a number of tool solutions proposed to solve the task problems. Each of the existing tools addresses a few features in the data-set, but no tool unites all the different aspects. In this study, the features of the previous tools are combined into a single tool to provide a better understanding of the data-set.
CHAPTER 1. INTRODUCTION

1.3 Problem

I use a synthetic data-set from the VAST Challenge 2010, Mini Challenge 3 created by Konecni [20]. The VAST challenge goal is to bring VA and InfoVis researchers to solve challenge problems and to evaluate their tool, using benchmark data-sets [14]. The VAST Challenge 2010 consisted of three Mini Challenges each with different tasks and a Grand Challenge that could be solved by combining the results of all three Challenges.

My research team (SFU-SIAT-IMAS) \(^1\) participated in the Mini Challenges and the Grand Challenge. We solved the Mini Challenge 3 tasks using IMAS \(^3\) tool to analyze the genomics data to determine the evolution of the Drafa virus. Our Grand Challenge submission won the best student analysis award \(^9\), and we were able to understand our tool’s strengths and pitfalls in solving the problems. For the purpose of this study, the data-set and the tasks for this study are modified from the original VAST Challenge. The reasons for this modification can be found in 2.1.3. The original VAST 2010 Mini Challenge 3 can be found in Appendix A.

The synthesized data (computer and human generated) of the VAST Mini Challenge 3 is about an illegal arms dealing scenario\(^2\) in which one of the dealers called Nicolai died in a hospital with symptoms consistent with Drafa Fever. In order to develop pandemic response plans, public health organizations need to get more information about the disease. Health professionals are seeking to help investigators use Nicolai’s health information to discover his contacts. Health officials have access to the genetic information of 56 patients (including Nicolai) that developed similar symptoms. They know that as the Drafa virus spreads from host to host, it mutates and evolves. For the given scenario, diagnostic tests confirm that the current outbreak is an evolved viral form (i.e. mutant strain) of the Drafa virus with a number of bases that modified over time. The first two questions are about relating the disease characteristics to new viral strains. The first Challenge task consists of questions about the distance between strains.

The data-set consists of 56 strains of a particular original virus, which are the result of spreading of a disease over time to different infected people. Each of these strains has a gene sequence of 1400 nucleotides with one or more nucleotide changes from the original virus’s sequence. Also, the genetic sequence of one strain that is considered to be the parent of all

\(^2\)http://hcil.cs.umd.edu/localphp/hcil/vast10/index.php/taskdesc/index
the strains is provided in the data.

The data-set contains the different viral strains genetic sequence with an identification label. The genetic sequences consist of coded nucleotides, which are A (Adenine), T (Thymine), C (Cytosine), and G (Guanine). Example of one viral strain sequence is as follow:

Label 118:
ATGTCACCGCCCTGCGCAGTTCATAGGGCCTCTCTGCCTCGGAACACGGGTCCTTTCC

There are some characteristics for each of the evolved viral strains and an explanation about the table. This table consists information for 56 strains. Table 1.1 shows disease characteristics data for few of the sequences.

Table 1.1: Sequence Characteristics Table. Definitions: Symptoms are what a patient experiences (e.g., pain, sore throat, vomiting, swelling, tremors). Mortality is a number of deaths as a result of disease. Complications is unfavourable evolution of illness (e.g. deafness, spontaneous abortion). Drug Resistance is mutant vulnerability to anti viral drugs. At Risk Vulnerability is disproportional effect on certain risk groups (e.g. children, elderly).

<table>
<thead>
<tr>
<th>ID</th>
<th>Symptoms</th>
<th>Mortality</th>
<th>Complications</th>
<th>Drug Resistance</th>
<th>At-Risk Vulnerability</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>Mild</td>
<td>High</td>
<td>Minor</td>
<td>Resistant</td>
<td>High</td>
</tr>
<tr>
<td>256</td>
<td>Severe</td>
<td>Medium</td>
<td>Major</td>
<td>Susceptible</td>
<td>Medium</td>
</tr>
<tr>
<td>19</td>
<td>Severe</td>
<td>Low</td>
<td>Major</td>
<td>Intermediate</td>
<td>Low</td>
</tr>
<tr>
<td>4</td>
<td>Moderate</td>
<td>High</td>
<td>Minor</td>
<td>Resistant</td>
<td>High</td>
</tr>
<tr>
<td>200</td>
<td>Mild</td>
<td>High</td>
<td>Major</td>
<td>Resistant</td>
<td>Low</td>
</tr>
</tbody>
</table>

1.3.1 Tasks

- Task 1: Identify mutations that lead to an increase in symptom severity (a disease characteristic). Each mutation provides the base substitutions and their position in the sequence, where the base substitutions occurs. Report findings in the order of importance. For example,

1. C → G, 456 (C changed to G at position 456);
2. G → A, 513 and T → A, 907 (G changed to A at position 513, and T changed to A at position 907);
CHAPTER 1. INTRODUCTION

Note that some of the mutations might have equal impact on a disease characteristic. Also, in some cases, a combination of the mutations explains a characteristic’s severity. These mutations should be reported together (e.g. second item above).

- Task 2: Identify mutations that lead to the most dangerous viral strains.
- Task 3: Nicolai has a strain identified by sequence 583. One patient has a strain identified by sequence 123 and the other has a strain identified by sequence 51. Which patient contracted the illness from Nicolai and why?

1.4 Approach

In this work, I select a benchmark data-set/task-set introduced in IEEE VAST Challenge 2010, and I assume that it is a valid task for the real users. During this thesis, I describe transformation of the Bioinformatics tasks to visualization tasks, design a visualization tool using the design literature in visualization field \( [13, 23, 40, 47] \), and evaluate the design using study design guidelines \( [6, 21, 27, 37, 38] \).

My goal is to design a tool to facilitate understanding and sense-making of the data-set/task-set for the real users. To update the design using target domain’s advices, I apply the User Experience Scenario evaluation \( [21] \) to identify effectiveness of the design for Bioinformatics users. Furthermore, I characterize this work as a design study in the visualization literature \( [38] \). Sedlmair et al. define the design study as: “design study is a project in which visualization researchers analyze a specific real-world problem faced by domain experts, design a visualization system that supports solving this problem, validate the design, and reflect about lessons learned in order to refine visualization design guidelines.” \( [38] \). They also introduce a taxonomy of design studies as follow:

**First type** Problem characterization and abstraction, to provide shared understanding between designers and domain experts.

**Second type** A validated visualization design. This type focuses on the visualization tool that is usually built for testing the ideas. There must be evidence to show that the tool is helpful for experts.

**Third type** The reflection on the design study that compares it to other related works \( [38] \).
CHAPTER 1. INTRODUCTION

The current work is consists of all the three types of design studies with different contributions in each. More specifically, this design study has contributed more on the second type and did not compare the outcome tool to other related ones.

Part of my contribution is an extensive review of the previous successful tools that are designed or used to tackle the same data-set/task-set. I critique each of the designs using InfoVis design literature [13, 23, 40, 47, 51]. I consider the produced knowledge from each of the previous related tools [4, 9, 12, 22, 29, 45, 50] to design the study prototype. One of the critical contributions of this work is several filtering options for users to exclude irrelevant data to solve the study tasks. The filtering options allow the user to work with a smaller set of data. Also, to the best of my knowledge having two modes for the system (Row and Column mode), as well as the Main View slider-bar is completely novel.

The research prototype is evaluated with visualization experts and is refined many times. Furthermore, I evaluate the final prototype with participants. The analysis of results enable me to make suggestions for future directions.

1.5 Thesis Outline

Chapter 2 is devoted to an overview of the relevant technical background and literature review. It follows with a broad discussion of the related work and how I position my work among them.

Chapter 3 provides a detailed discussion on the contributions to the first part of the methodology that is the design process. This chapter includes discussion on the design guidelines, options for each design and the current design justification. This chapter continues with the implementation snapshots and sample usage scenarios.

Chapter 4 contributes to the second part of the methodology, i.e. the evaluation of the design for helping the iterative nature of this work. The design of the study comes in this chapter.

Chapter 5 contains the results of the study and analysis of the results.

Chapter 6 focuses on the detailed discussions of the strengths and weaknesses of the current tool by analyzing the implication of the evaluation results of the chapter 5. This chapter presents the author suggestions of design options for the future directions.
Chapter 2

Literature Review

The aim of this chapter is to provide a review of the literature regarding the problem that I address throughout this research. The first section of this chapter presents the necessary Bioinformatics background information required for understanding the research problem domain. The second section covers reviews of the visualization techniques that are used to design the study research, their advantages and limitations.

2.1 Technical Background

This section provides the biological context of the current problem under investigation, as well as detailed descriptions of the tasks and the input data used for this research.

2.1.1 Biological Data Visualization

Applying the principles of Information Visualization to design biology-specific systems is highly acknowledged in the literature [18, 25, 26, 30, 32]. Advanced hand-drawn pictures in scientific publications prior to existing computers shows that utilizing the human vision system was grounded in biology many years ago [32]. As the biological data-sets scales are increasing rapidly, custom software combined with manual intervention is replacing manual data analysis in biological sciences [18]. These computer-based visualization tools have enhanced our ability to communicate with the large amount of scientific data. Usually these tools are designed for a specific data-set/task-set in the domain. Advantages of these custom tools are twofold. First, they solve target analysts’ problems, which are part of the domain
problems. Second, by analyzing the successful tools, researchers can eventually extract the target domain’s design guidelines and patterns. A special issue of Nature Methods gives biologists an overview of current computational methods and tools used for visualizing biological data [32]. Although classical visualization techniques are used in the field of biology, researchers define new and creative ways to meet the target domain visualization needs [15]. One such example is the work of Nielsen et al. in creating a novel graph representation for visualizing genome sequence assembly structures [30].

In the next section I describe the necessary biological background required to fully understand the context of FilooT.

2.1.2 Biological background

Viruses cannot replicate or evolve without the use of a living-cells machinery. Once a virus infects a host, it makes copies of itself, growing the population of virus within the same host and eventually spreading to other people. During the viral replication process, its gene sequence has to copy and transmit the exact same sequence (between 7000 to 500,000 nucleotide bases) to its child cells. During this process, typically some mistakes can be made and as a result, some changes appear in genetic sequence [31]. A genome sequence consists of single nucleotides that coded as A (for adenine), T (for thymine), C (for cytosine), and G (for guanine) [19]. Each substitution replaces an existing nucleotide in the gene sequence with a different one (e.g. A changing to C).

One way for characterizing DNA is to compare their sequences with each other [19]. In bioinformatics, aligning the sequences in rows helps finding the similar regions between them. In the case of having Pairwise Alignment, analyzers compare the sequence of one gene to the sequence of another; that, in many cases, is useful for their tasks. In the case of having a set of DNA sequences however, comparing each sequence against all the others in the set is not only time consuming, but also leads to inaccurate results. The main reason is that two sequences are compared without considering the knowledge of other similar DNA sequences. In order to compare more than a pair of sequences, all sequences must be compared to each other in a heuristic optimization process called Multiple Sequence Alignment. Multiple Sequence Alignment of a family of DNA sequences represents a more general model of them. This general model is called Sequence Profile.

A Sequence Profile has information about all the sequences in each position. Figure 2.1 shows the concepts of Alignment Matrix, Profile Matrix and the Consensus Sequence. The
CHAPTER 2. LITERATURE REVIEW

Figure 2.1: An example of the multiple sequence Alignment Profile Matrix. Seven sequences aligned in the Alignment Matrix. Profile Matrix contains information about each base in each position across all the sequences. The Consensus Sequence contains the most probable base in each position [19], © MIT press 2004.

Profile Matrix calculates the number of different bases in each column, and the Consensus Sequence consists of the most dominant nucleotide in DNA sequences in each position across all the sequences [19]. Considering a family of sequences, there are two types of positions across the entire multiple sequence alignment. The first type is a variable position in which mutations are allowed, meaning that these changes will not cause a loss of function in the sequence. In the contrast, if a substitution occurs in the second type positions, a loss of functionality happens. In the case of virus families, some changes in genetic sequence in some locations could strengthen or weaken the virus and result in a more or less dangerous disease.

In the selected data-set and task for this research, health investigators are interested in finding the relation of the gene substitutions to disease characteristics in a virus gene family of 56 sequences. The virus causes a disease called Drafa. The Drafa disease and the DNA sequences are artificial. Also, the VAST Challenge explicitly stated that the DNA is to be treated as a non-coding, so that Codon analysis and AA sequence analysis is not to be considered. For solving the VAST Challenge problem, visualizing the Multiple Sequence Alignment result is helpful. Using FilooT’s Multiple Sequence Alignment visualization (Main View), the analyzer detects the substitutions in evolved viral strains that
make a virus cause more severe disease.

There are generally two types of visualizations for Multiple Sequence Alignment: Multiple Sequence Alignment viewers and the Sequence Logo [36]. In the following sections I discuss some of the methods of Multiple Sequence Alignment viewers. The Sequence Logo visualizations is useful in the case that the probability of a base in a location or the weight of a base in the model is important. In this research, Sequence Logo visualization is not relevant for the task problems and it is not used in the design of FilooT.

2.1.3 The data-set/task-set

The data-set/task-set for this study that is introduced in Section 1.3.1, is a modified version of the original data-set/task-set. The original data-set is described in Appendix A. Following section discusses different approaches to tackle this data-set and also contains the reasoning behind the modification.

In the original challenge, in addition to information regarding the 56 Drafa virus outbreaks, there was also information about the outbreak of this virus in different countries. Finding the county of origin of all 56 outbreaks was of the VAST Challenges original tasks. To answer this question, my research team assumed that the native sequence which displayed the most similarities to each of the current outbreak sequences was the ancestor of all the outbreak sequences. We defined the similarity between two sequences as the number of different bases; those sequences that had the least number of base differences were the most similar. We aligned each current outbreak sequences against all the countries to detect the country with the most similarity to that sequence. Figure 2.2 shows a segment of the Pairwise Alignment (see Section 2.1.2) results for sequence 118 against all the countries. The light purple areas show the similar regions and the green rectangles highlight the differences (ignore green colour gradients). The first sequence is Nigeria_B, which has at least three substitutions, and because this sequence has the fewest highlighted areas, it is the most similar to sequence 118. We repeated the same steps for all of the outbreak sequences, and observed that Nigeria_B strain were the most similar sequence to all of the current outbreaks.

Another question in the original VAST challenge was about discovering Nicolai’s contacts. The arm dealing investigators suspected two of the patients were Nicolai’s contact. They wanted to know which one was the most likely to contract the disease from Nicolai. To answer this question, our team assumed that the patient with the most similar sequence
Figure 2.2: SFU-SIAT-IMAS team submission, IMAS snapshot solution for task 1. The sequence 118 is aligned against each of the counties in each row. The first country contains fewer number of highlighted region, indicating that it is the most similar sequence to 118. Adapted from the previous created image by the author.

to the sequences provided in the challenge had the highest probability of contracting the illness from Nicolai. So our strategy of solving this question was very similar to the first one; we compared each suspicious patient’s strain’s to Nicoai’s by counting the highlighted areas in Pairwise Alignments in IMAS and reporting the one with fewer differences.

In addition to the stated solution, there is another way of addressing the two first questions that considers the evolutionary relationship of the virus outbreak family. Some challenge participants reconstructed the likely evolutionary paths using the distance between genetic sequences of other patients.

The evolutionary relationship reconstruction was addressed differently in the submitted solutions. Some submissions used computer software programs to generate the relationships of the 56 patient sequences. However, most software packages that reconstruct hierarchies
of genetic data, do not work well with single parent genetic sequences [4] as it is the case in this data-set. Therefore, Noblis team [4] created a pedigree tree in Microsoft Visio manually. Although the construction of the evolutionary tree in not clearly explained, I inferred that they must have used the information from those software programs as a guide to creating the relationship tree. Some of the submissions [22, 12] used the Minimum Spanning Tree for constructing the evolutionary tree. The weight of the edges was the Hamming distance between the two nodes [19]. The Hamming distance was the number of positions that differed in any two rows that implied the number of changes is needed to transform one sequence to the other. The Minimum Spanning Tree is a tree in a graph that connects...
CHAPTER 2. LITERATURE REVIEW

all nodes(rows) and its total edge weight is the minimum of total edge weights of all the possible trees. Figure 2.3 shows one visualization of the evolutionary tree submitted by [12].

The strategies for solving both above mentioned tasks in either method (pairwise alignment or using the evolutionary tree) are very similar to each other. Through my observations the two strategies could be considered related because one can use the information of the Pairwise Alignment to construct the evolutionary tree. Given the similarity of the solution strategies of the two above mentioned questions, I only selected one of them in this study (i.e. the second one). The genetic sequence of Nigeria_B is kept and used as a root of the outbreak evolutions. All the other county sequences are excluded from the original data-set.

In the yet another question of the original challenge, the health analyzers needed to understand how characteristics of the disease related to virus strain mutations. In particular, they were interested in changes in genetic sequence that made the symptoms more severe, and what substitutions make the most dangerous overall strains. These two questions are included in this study tasks and are detailed in Section 1.3.1.

2.2 Positioning This Study in Related Work

This section provides a discussion on a number of related tools and visualization techniques that are used in this study. Multiple Sequence Alignment Viewers and Matrix Viewers are common in the other tools similar to FilooT. This section continues with a discussion on graph and tree visualization techniques used in the design of Graph View (see Section 3.2.4.4) of this thesis study prototype.

In the design of FilooT, some of the generated views are manipulatable. In addition to the most common techniques for manipulating the views, which are scrolling and zooming, there are two other techniques that I discuss in this section: the Overview+Detail, and the Focus+Context [24]. This chapter is concluded with critiques on the chosen solutions from the VAST challenge 2010 submissions. The purpose of this review is to learn from their useful features and avoid their pitfalls.

2.2.1 Multiple Sequence Alignment Viewer

A number of tools in the biological data visualizations contain a variation of the Multiple Sequence Alignment view [32]. Multiple Sequence Alignment view is a table representation, in which each row corresponds to a sequence and each column is a position in all the
sequences. In the case of DNA sequences, each cell represents a DNA letter in each sequence. An important characterization of this view is to show the variations in the sequences to the analyst. Highlighting the differences in sequences is performed in three different ways:

- Using colour for each type of nucleotide. Figure 2.4 shows examples of nucleotides colour choices by a number of visualization tools [32].

![Figure 2.4: Different colour codes are used to encode A, C, T and G. Figure reprinted by permission from Macmillan Publishers Ltd [32], © Nature Publishing Group, 2010](image)

- Colouring some cells depending on two features. First their type of nucleotide and second, the weight of presence of that type of nucleotide in a particular position. This approach makes the pattern less confusing [32]. Figure 2.5 demonstrates an example of the shading method.

![Figure 2.5: The shading approach in multiple sequence alignment viewers. Figure reprinted by permission from Macmillan Publishers Ltd [32], © Nature Publishing Group, 2010](image)
CHAPTER 2. LITERATURE REVIEW

- Highlighting just the consensus in each position, which is a simplified version of shading [32].

Historically, the table view provides interactive features to allow the users to gain insight about the data [39, 48]. The following sections covers a number of these tools that primarily influenced my design.

IMAS (Interactive Multigenomic Analysis System)

IMAS [39] is a visual analysis tool for performing rapid analyses of DNA sequences. This tool visualizes the output of common bioinformatics tools such as BLAST program for Pairwise Alignment and Clustal-W for Multi-alignment in a unified framework. In order to align multiple sequences, the user selects a number of Pairwise Alignments that she/he initially made with IMAS. The Multi-alignment view in IMAS is zoomable and gives the user an overview of the entire data-set in a single view when it is fully zoomed out. To control the level of detail, the user can zoom in to a region of interest. The horizontal and vertical scrollbars enable the user to navigate through the data at all other zoom levels. The display updates in real-time, which makes the system efficient for the end users. Figure 2.6 shows different parts of the Multi-alignment view in IMAS.

One drawback of this view is lack of interactivity with the user. The user is not able to sort the rows. The ability to add metadata to a row is not predicted in this design. The rows’ positions are fixed and the user cannot change them. The user is also incapable of adding a row to or deleting a row from multi-aligned rows. The only way to change the rows in a multi-alignment is to make a new multi-alignment. This task involves a lot of selection of pairwise alignments and is not efficient.

Jalview (A Multiple Alignment Editor written in Java)

Jalview is one of the most commonly installed tools. The two papers describing Jalview have attracted over 1,100 citations with the most recent version [48]. The Multiple Alignment view is capable of handling the sequence hiding (row hiding). The user can select rows and right click on them to hide them. If the user hides some sequences, a symbol will appear to indicate some rows are hidden. If the user right clicks on that symbol, she/he can expose the sequences. The same feature applies to columns. In Figure 2.7(a) some of the rows and columns are hidden.
Another feature is that the user can select a number of sequences and make a group out of them. The group name will be shown when the user places the mouse over the sequences. The group making feature is supported only for sequences (rows) and not for positions (columns).

The sorting feature allows the user to sort the sequences with a number of different criteria, such as labels of groups. The sorting criteria is shown in Figure 2.7(b). One limitation I discovered is that the sorting criteria are predefined and the system does not support the user creating their own. For example, one sorting criteria that the user may want to create is sequence metadata. Another limitation is that the user can not move the rows to sort them manually.

Another useful feature of Jalview is that the user can change colour choices. For example in Figure 2.7(c), the right view shows the exact visualization as the left, but in a different
(a) Rows and columns are hidden under the blue arrow.

(b) Sort options.

(c) Two views are created and expanded side by side to ease the comparison. The second view uses Clustalx colour choice. Mouse over another view.

(d) The status-bar shows what is happening.

Figure 2.7: Jalview features. Figures are screenshots of Jalview [48].
colour. Figure 2.7(c) also reveals another valuable feature in Jalview which is the ability to create different views out of the same multi-alignment. A newly created view inherits some features from its root, such as hidden rows and columns, and user-defined groups. The views are shown in different tabs and the user can additionally split them into different windows for a side to side comparison. There are also small overview thumbnails of the other existing views in the interface that help with switching between the views. The two other views in Figure 2.7(c) are shown in small overviews in the bottom-left side of the page. Placing the mouse over these thumbnails shows their name. Although this thumbnail feature is useful, there is no map which displays the relationships between the views, nor the order they were created.

When the user’s mouse hovers a sequence ID, the system highlights the entire sequence in the row. The same feature is available for a column ID. This feature is very helpful in highlighting a row or column, but the default highlighting colour (red) covers the underlying information in the row and the column.

Finally, Figure 2.7(d) shows that embedding a status-bar in the design aids the user to estimate the remaining time.

2.2.2 Matrix Viewer

The multiple alignment views usually accompanied by another metadata matrix view are a kind of Table Lens where each column contains information about one metadata and each row represents the value of that metadata for each strain. The rows are sortable according to the values of each column [33]. iHat is an example of a table representation for finding mutations [45].

Freire et al. [11] used horizontal bars (the Length visual channel) for encoding each cells’ data, as well as vertical bars on top of each column to show overall column distributions. Others, such as Sopan et al. [43], used colour saturations in different column cells to encode their values. The colour saturation usually is a better choice for visualizing the information in a cell as it has higher accuracy for encoding ordered data [23].

2.2.3 Hierarchy representations

Some data has a hierarchical structure. One common visualization technique for representing this structure in the data is tree (or a special node-link diagram) visualization [15].
The tree visualization could have different layouts. We could use Polar or Cartesian coordinates to show the tree structure. Heer et al. categorizes the types of tree layouts as the following [15]:

**Node-Link Diagrams:** Nodes are used to represent each entity, and usually the relationships between them are represented by lines or spaces. Figure 2.8 includes an example of cartesian node-link diagrams 2.8(a)), radial node-link diagrams (2.8(b)), and indented diagram(figure 2.8(c)).

**Adjacency Diagrams:** In this diagram, nodes are areas and their relative position shows their relations. Figure 2.9 includes an example of the icicle layout (figure 2.9(a)), and a polar version of it that is called the sunburst layout (Figure 2.9(b)).

**Enclosure Diagrams:** These graphs use containment rather than adjacency to show the
CHAPTER 2. LITERATURE REVIEW

(a) Icicle Layout

Figure 2.9: Types of Adjacency Diagrams.\cite{15}, Figure reprinted by permission. © 2010 ACM, Inc. http://doi.acm.org.proxy.lib.sfu.ca/10.1145/1794514.1805128.

(b) Sunburst layout

(a) treeMap

Figure 2.10: Types of Enclosure Diagrams.\cite{15} Figure reprinted by permission. © 2010 ACM, Inc. http://doi.acm.org.proxy.lib.sfu.ca/10.1145/1794514.1805128.

(b) Circle layout
CHAPTER 2. LITERATURE REVIEW

Figure 2.11: The tree visualization from Noblis team submission for the VAST challenge 2010, mini challenge 3. The diagram uses the sunburst layout to show the evolutionary relationship between the sequences. The colours represent the degree of the overall danger level of the sequences. Figure reproduced [5] with permission from Catherine E. Campbell.

tree structure [15]. Figure 2.10(a), shows tree-map layout, and circle-packing layout is shown in Figure 2.10(b).

This taxonomy could be helpful for understanding different tree visualizations. I would argue that the indented layout could go to the Adjacency Diagrams category because the relationship is shown using the spaces between the nodes. The nodes in one category are in the same horizontal position close to each other and the space between categories is related to their hierarchies.

In this study’s data-set, there is a hierarchical relationship between the strains. The data-set does not provide this relationship but it is implicit in the definition of the data. In the VAST challenge solutions, The Noblis Team [4] used sunburst layout to represent the evolutionary tree of the current outbreak sequences. Figure 2.11 shows this layout.

In another example, Freire et al. [11] used the basic Node-Link layout for the evolutionary tree information. Figure 2.3 shows the evolutionary tree. In FilooT, I adopted the same representation.
2.2.4 Network representations

In some cases there are other types of relationships between data items, which could be considered a general form of hierarchal relationship. The graph representation is commonly used to show other kinds of relationships. Figure 2.12 shows two examples of graph layouts. In Figure 2.12(a), the nodes are connected directly with links. This type of representation comes in different layout algorithms in order to find proper locations for the nodes to minimize edge crossings. The second example 2.12(b) is called adjacency matrix. The row and columns represents different nodes and the coloured area indicates a link between nodes. This type of representation usually requires an efficient sorting algorithm since by changing the order of the rows (and columns), different patterns could be revealed.

The two above graph representations are the design alternatives for encoding the relationship between columns (position) in this study.

2.2.5 Overview+Detail

The Overview+Detail shows different detail levels in different frames that are separated by time or space [7].
Separated by time: This feature enables the user to observe multiple levels of detail for data in the same space in different times. The semantic zooming navigation in IMAS [39] provides this kind of features to the users. Zooming-out of IMAS Multi-alignment view, the genetic letters disappears and only simple rectangles remain after a certain level. The opposite occurs for the zooming in. This interaction assists the user in discovering patterns in the whole sequence area by controlling the level of detail. Those patterns could emphasize non-conserved regions at a glance. The user then could zoom-in to a region of interest and observe more detail.

Separated by space: This technique is used when the user wants to see the overall structure of the content while at the same time working with full detail data. Group View in FilooT has a thumbnail for each created group that enables the user to observe an overview of the group data without a need to zoom in Main View. Due to the physical separation of the two views, users interact with the views separately, although actions in one are immediately reflected in the other.

2.2.6 Focus+Context

This technique differs from Overview+Detail technique by providing the detail and the surrounding context simultaneously. TreeJuxtaposer [28] is an example of using Focus+Context technique for navigation in a way that visibility is also guaranteed. Sequence-Juxtaposer [42] is another example of applying Focus+Context in bioinformatic sequences alignment explorations. Although this technique could have been useful in FilooT Main View, because of the time limitation, it is not incorporated for FilooT and it is suggested as a future work.

2.2.7 Other VAST Challenge Solutions

In this section, I provide a thorough review and critique of the four winners of the VAST Mini Challenge 3 in order to use the produced knowledge for my design. The VAST Challenge evaluators commented on a number of analytic processes as well as visualization issues [37]. As a result of the evaluators’ careful reviews, all the winner tools could be considered as examples of good overall solutions for the challenge.

The evaluators’ criteria includes complexity of the visualization, colour choices, intuitive symbols, amount of manual and repetitive steps in the process, number of levels of menu
and options to check [37].

Below is the evaluations of each submission in detail.

**Submission Title: Gene similarity uncovers mutation path**

**Award Title:** Innovative tool adaptation  
**Group Name:** ManyNets submission  
**Group Affiliation:** Universidad Autnoma de Madrid and University of Maryland

Freire *et al.* [12] adapted ManyNets [11], a network visualization tool with tabular interface. The tabular view was a kind of Table Lens that the disease characteristics were shown in columns. There was a feature of creating a new column with the existing characteristics. Clicking on a column header sorted all the rows according to the values of the particular characteristics associated to the column.

Generally, ManyNets is used to discover many networks, but in the challenge there was only one network with nodes that are associated with the rows (sequences) and edges that are the number of mutations needed to be changed to reach from one node to the other. With the help of tabular view, the researchers founded a tree rooted at the node 531 in which any direct neighbour of a node had only one different nucleotide (hamming distance of one). The only exception for this rule was node 961. Figure 2.3 shows how they visualized their tree using simple node-link layout. Using their tree visualization, they discovered the responsible paths for the severe mutations. In Figure 2.3, the nodes are coloured on overall disease characteristics. The three highlighted orange nodes are the most severe ones. To report the results, they traced the path of the founded nodes to the root looking at the Table Lens view that explicitly showed the mutations. Figure 2.13 shows the mutations of the original node to nodes 118, 123, 501.

![Figure 2.13: ManyNets tabular view. Figure reprinted with permission from Manuel Freire [12], © 2010, IEEE.](image-url)
Submission Title: Cognizant BFS Innovations prototype

Award Title: Innovative Visualization

Group Name: Cognizant BFS Innovations rapid-prototypes submission

In order to get insight about the data, Nene et al. [29] built a tool that helped them identifying mutations that worsen disease characteristics.

Their visualization method is shown in Figure 2.14. Three of the columns, columns 842, 790, and 946 are shown in this snapshot. Each column splits into two groups. The top group is for dominant nucleotide in a position across all the sequences, and the bottom region represents changed nucleotide in that particular position. The nucleotides colour code is as follows: orange for A, blue for T, green for G, and purple for C.

The disease characteristics is the row of five cells on top and the bottom of each column. The symbol S for symptoms, M for mortality, C for complications, D for drug resistance, and V for at-risk-vulnerability. There are two numbers associated with each characteristics, one is the dominant nucleotide (the upper group) and the other one is for the mutant nucleotide (the lower group). The numbers are the percentages of each group that are severe in that particular characteristics. For each characteristics, the higher number among the two groups is coloured in red and the lower is coloured in orange.

There is a third number at the button of each characteristics in a column that is shown in white in the black background area. This number represents the absolute difference between the percentages of the two groups. If the mutant group (the lower group) disease characteristics colour is red, then some bars are coding the absolute numbers as well. The blue bars are guidance for understanding the relationship between mutations and symptom severity. The circles are summation of the bar numbers, and the pink circles are guidance to finding the mutations led to overall disease severity. The following example further illustrates their interface.

Considering position 790 in Figure 2.14, the interface shows that 44 of 58 sequences have letter T in that column, and 14 of them have letter C. Assuming that the higher number of nucleotides are dominant, letter T groups are at the top showing with blue, and C is considered mutation at the bottom group with purple. Twenty two% of the top group have severe symptoms (‘S’), while 28% have non-severe rows in them. Thus, the 28% is greater and it is in red at the bottom of the column in S cells. Because this greater number is among the mutant group, the bar for the absolute value which is (28-22=6) is shown. It
is coloured blue to help finding the mutations that led to severe symptoms. In order to
find the mutations led to overall most dangerous sequences, one can look at the summation
value shown in the circle (6+85+33=124). The greater blue bar in column 842 indicates
the relationship between this column’s mutation and symptom severity. Additionally, the
column 946’s circle value indicate its its impact on overall danger.

Figure 2.14: Solution given by team Cognizant BFS Innovations. Three of the columns,
842, 790, and 946 are shown in this capture. Figure reprinted with permission from Har-
shawardhan Nene [29], © 2010, IEEE.

There is no doubt that the design was innovative, although it involved a higher learning
curve. Even though the statistics of each group automatically generated and shown in a
separate block helped understanding and comparing the characteristics of each group, I argue that the fully automatic solution does not allow user customization and discovery. If the user only needs to look at the result and pick the answer, perhaps there is no need for a visualization tool [40]. The other issue is that the percentage measure is not accurate enough. In the design of FilooT, I use this concept with an statistical measure in P-value View. For the visualization, using the hue visual channel is a good choice to differentiate between categories, but it is not strong enough when some colours such as blue, orange and red are used for different purposes in a single interface.

Submission Title: GeneTracer: Gene Sequence Analysis of Disease Mutations

Award Title: Excellent Process Explanation

Group Name: GeneTracer submission

Group Affiliation: Georgia Institute of Technology

GeneTracer [22] was developed for the VAST 2010 genetic sequence Mini Challenge. It had three views: Gene Sequence view, Disease Characteristic view, and Graph view. The GeneSequence view was a multiple alignment view that the colour of each cell indicated its base: A (red), T (green), C (purple), G (blue). To reduce redundancy, they built a button to remove all those columns that contain the same gene bases across all the rows. In the design of FilooT, I have developed this idea further and built a filter that removes the columns on different levels of substitution numbers (which here only is zero). Figure 2.15 shows the Reduce Redundancy button. The blue horizontal lines separate the different levels of a disease characteristic among the sequences.

Additionally, the row order could be changed which was helpful to reveal the pattern between the rows. The comparison between rows was manual; first the user put the selected rows beside each other, and then compared them visually. In order to guide the comparison task, the researchers built an additional column with the saturations of brown colour that indicated similarity measure between all the rows and the selected one. The lighter brown showed that they were more similar.

In Disease Characteristic view, they had a Table Lens where each column had a different colour and each cell had different saturations of that colour. Darker means more severe, and the horizontal rows indicated severity levels. The researchers placed additional cell on top of each position that indicated variations across regions within that column. Darker meant higher variance.
The Graph View visualized the relations among the sequences via a Minimum Spanning Tree representation where the weight of an edge between two sequences is the Hamming distance between the two [19].

**Submission Title: SequenceView**

- Award Title: Good overall design and analysis
- Group Name: giCentre - sequenceView submission,
- Group Affiliation: City University London

Wood *et al.* [50] provided a button to hide common regions across all sequences. This feature was similar to the GeneTracer redundancy button. Their Table Lens representation utilized a heat map. An interesting feature in their Table Lens was two levels of sorting of the rows (Sorting the rows according to one characteristic, then sorting them again within each category of the first sort, according to the second characteristic).

Another useful feature in their submission was the idea of the complement relationship.
In the next chapter, I cover the idea about a relationship between different columns in full detail.
Chapter 3

Methodology: Design

This chapter covers the technical background on Information Visualization tool design. In this section, I justify my designs with the described technical background. The chapter is followed by a discussion on the implementation and concludes with screenshots of the system. Two scenarios of using the system to solve the study tasks are demonstrated at the end of the chapter.

3.1 Technical Design Background

This section shows some knowledge and understanding of my view on Information Visualization. I use this section’s ideas to design the study’s tool.

3.1.1 Information Visualization

Each visual encoding consists of a set of marks. The mark set includes the rectangle and circle, text, Gantt bar, line, polygon, and image. Bertin suggests that the visual properties of the marks (also called visual channels) are shape, size, orientation, colour, and texture. The design guidelines are mainly about how to best use the marks and channels to represent the desired information for the target users. The encode relation is the relation between graphical objects and their properties and the information that they represent. In this relation, graphical objects are points, lines and areas and the properties of the objects are position, colour (hue and saturation), shape, size, orientation, texture, length. Two principle guidelines in InfoVis design literature...
for the encode relation are expressiveness and effectiveness suggested by Mackinlay [23].

The expressiveness criterion is to make sure the designer includes all the important features of the data in their design. Furthermore, the designers must avoid adding any extra facts that do not exist in the data.

The effectiveness guideline ranks the visual attributes according to their effectiveness to be used for different types of data. According to Ware [47], InfoVis utilizes three types of data: categorical (or nominal), ordinal (or integer data that is discrete and ordered), and real number data. The rankings suggest that the higher visual properties are more effective for encoding the selected type of data. The ranking guides also suggest the principle of importance ordering [23], that is to map the most effective channels for the most important features that to be represented. Figure 3.1 shows one such list suggested by Mackinlay [23].

![Figure 3.1: Mackinlay's ranking list of effectiveness of visual properties of the mark depending on the data type. In each column, the effectiveness of the visual attributes for encoding each of the different types of data decrease from top to bottom.](http://doi.acm.org.proxy.lib.sfu.ca/10.1145/22949.22950)

Such rankings are suggested by researchers based on their experience and, to the best
of my knowledge, have not been theoretically proven. However, the rankings are evidenced by the knowledge of human perception and the experimental evidence in the literature. For example, the core idea behind the ranking list of Mackinlay is its consistency with “Accuracy” [23]. Accuracy means that the quality of a visual representation depends on the human perception capability, i.e. humans do not perceive different visual channels equally.

Throughout the design of FilooT, I utilize the following rules based on the above ranking:

- The position property is carefully considered because it is very effective in visual encoding (in all the views).
- Colour hue property is used for encoding an important nominal (categorical) channel (in Main View).
- Length property is used for encoding an important qualitative information with bar lengths (in P-value View).
- Colour saturations is preferred over length property for encoding ordinal data (Matrix View).

Although the ranking lists are helpful resources for picking visual channels to encode different types of data, the encoding design is not always a simple and straightforward task. Specifically for designing more complex encodings, some other principles such as other InfoVis and perceptual guidelines should be kept in mind. The following list contains some other InfoVis and perceptual guidelines that I follow while designing FilooT:

- The dynamic range: The number of dynamic range that we could use to match the steps of an attribute. An example of this is the orientation channel. Each step in the orientation needs to be at least 30 degrees [44].

- Integral-Separable dimensions: For separable dimensions, the attributes are perceived independently, whereas in integral dimensions, the user does not distinguish between different attributes but observes them as a whole. For example, width and height in a rectangle is perceived as a size of rectangle. It is not seen as two separate dimensions [47]. (The combination of the attributes in channels makes a new attribute and they would lose their primary property)
Integral-Separable dimension pairs: The visual channels are dependent to each other and it could change our perception when they are used together. For example, our judgement of an object’s size could be different when it comes in different colours [47].

Pre-attentive Processing: The fact that some of the channels could make some object more distinguishable from its surrounding. For example, curved lines can be pre-attentively distinguished from straight lines [47]. This could help visualize things that pops up to people “at a glance” [47]. Highlighting objects with a colour that is distinguishable from their surrounding is another example and a common interaction in InfoVis designs.

Monotonicity of Visual Attributes: The fact that some channels have the intuitive meaning of greater or less and some do not have. For example colour saturation.

Also there are some rules that are beneficial for organizing data into groups [47]:

- Proximity: spatial proximity indicates the organization of objects.
- Similarity: objects with similar shapes are perceived to be in the same group.
- Connectedness: line connectors are powerful to show relationship between objects. For example, objects are connected by lines in a node-link graph.
- Symmetry: group similarities are perceived more easily if the objects are organized symmetrically.
- Closure: closed contours are effective to show inside, outside, and overlapping relationships of segmented regions.

3.1.1.1 Interaction Techniques in Visualization

A fundamental part of a visualization is interacting with the user, meaning that pictures are not static, but could be changed upon user request. There are several types of interactions, such as filtering, sorting, selecting, navigation, and changing the visualization. One guideline for design changes is using animated transitions [17, 34] that provide smooth changes. Yet, choosing to have a change needs to be considered carefully because in many situations having different views or information side by side is more effective for the analysis tasks [17]. In this regard, one important consideration in design decisions is about making
graphics in single view or linked-multiple views. There is also some kind of interactions that are defined only for multiple views. One powerful example of such interactions is Brushing and Linking. FilooT has four different views each one reveals a different feature of the selected data. There is a linking interaction between the views and some of the changes are designed to be smooth with the help of animation.

Aggregation methods could also be considered as a kind of interaction in which users combine and summarize the information. The aggregated variable could be shown as added information to the data, or be replaced with the original data for some purposes such as reducing the data. A very common form of aggregation is when the visualization generates statistical information about data such as average, min, max and so on. In the Matrix view of FilooT, the user could add new variables with aggregating the existing disease characteristics using simple operations such as sum.

### 3.2 Applying Technical Design Background to FilooT Design

This section covers the design ideas for building the tool of this research called FilooT. The early paper prototypes in the form of sketches were used for describing the design ideas.

The design prototypes and the implementation are almost identical except for some of the ideas that were not implemented for the study and remain mockup. This section clarifies whether the idea are implemented or not.

The first step towards the design of the system is to free up my mind as a designer, from the biology domain concepts. In order to characterize the problem with the visualization domain language, I map the vocabulary of the task-set/data-set of Bioinformatics domain to the InfoVis vocabulary data-types/tasks. The result of this step is used later to design visual encodings/interactions [27].

The translation step could result in two levels of task abstraction. The high-level translation and the low-level version with more details. According to Amar et al. [1], examples of higher level tasks are decision-making, especially under uncertainty, learning a domain, understanding trends, and predicting the future. They also suggested examples of low-level analytic tasks such as filter, sort, determine range, find anomalies, correlate, cluster, and compute derived value, retrieve value, and characterize distribution. The high-level tasks could be recognized and captured when people are interacting with the InfoVis tool to understand the data. The low-level tasks are the result of breaking the high-level tasks in
steps. Having these task levels in mind helped me design the tool step-by-step based on the identification of different visualization tasks.

In order to design the visual encodings and interactions of my prototype to support those InfoVis data-types/tasks, I translated the original domain and tasks to InfoVis data-types and tasks.

### 3.2.1 Domain Data Translation

The data translation includes characterizing the data-types and derived variables. Our domain data-set consists of two tables. Table 3.1 contains 56 virus strains, each of which is in the order of 1400 nucleotides in length. The second Table 3.2 contains disease characteristics metadata for each strain. The disease characteristics table has 56 rows corresponding to the virus strains and six disease characteristics as metadata.

**Table 3.1:** A sample part of the virus strain sequences table.

<table>
<thead>
<tr>
<th>Sequence ID</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>118</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>G</td>
<td>A</td>
<td>C</td>
<td>T</td>
<td>C</td>
</tr>
<tr>
<td>200</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>G</td>
<td>C</td>
<td>G</td>
<td>T</td>
<td>A</td>
</tr>
<tr>
<td>19</td>
<td>A</td>
<td>T</td>
<td>G</td>
<td>G</td>
<td>C</td>
<td>C</td>
<td>T</td>
<td>C</td>
</tr>
</tbody>
</table>

**Table 3.2:** A sample part of the sequence characteristics table. There are six disease characteristics in the data-set and this table contains four of them.

<table>
<thead>
<tr>
<th>Sequence ID</th>
<th>Symptoms</th>
<th>Mortality</th>
<th>Complications</th>
<th>Drug Resistance</th>
</tr>
</thead>
<tbody>
<tr>
<td>118</td>
<td>Mild</td>
<td>High</td>
<td>Minor</td>
<td>Resistant</td>
</tr>
<tr>
<td>200</td>
<td>Severe</td>
<td>Medium</td>
<td>Major</td>
<td>Susceptible</td>
</tr>
<tr>
<td>19</td>
<td>Severe</td>
<td>Low</td>
<td>Major</td>
<td>Intermediate</td>
</tr>
</tbody>
</table>

### 3.2.1.1 Translating the DNA Information Table

Table 3.3 shows an example of transforming Table 3.1 to what is suitable for visual encoding in this project. The translation is done as follow:

“Row”s represent the strain sequences, while “Column”s are the positions in aligned sequences. Nominal data translated to a derived cell variable are written in cell attributes. “Derived Cell Variable” indicates whether there is a substitution in the sequence/position in comparison with the original sequence. The original sequence comes in the first row.
Table 3.3: Translated sample part of the virus strain sequences table.

<table>
<thead>
<tr>
<th>Sequence ID</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>118</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>G</td>
<td>A</td>
<td>C</td>
<td>T</td>
<td>C</td>
</tr>
<tr>
<td>200</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>19</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

3.2.1.2 Translating the Table of Disease Characteristics

Table 3.4 shows an example of transforming Table 3.2 to another data that is suitable for visual encoding in this project. The translation is done as follows. Strain characteristics information are shown in each “Row”. Each “Column” contains one characteristic. A ”Cell Attribute” represents an ordinal variable, which is translated to numbers. The translation is an example of many possible ways of turning the ordinal data into numbers. Also the distances between the values are not necessarily equal. “Derived Attribute” (Overall) is an aggregated characteristics that is added to the list of metadata. This attribute is the summation of all the characteristics to represent the overall danger level as it will be needed in the task problems.

Table 3.4: Translated sample part of the sequence characteristics table.

<table>
<thead>
<tr>
<th>Sequence ID</th>
<th>Symptoms</th>
<th>Mortality</th>
<th>Complications</th>
<th>Drug Resistance</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>118</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>200</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>19</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>9</td>
</tr>
</tbody>
</table>

3.2.2 Domain Data Visual Encoding

Figure 3.2 shows visual encoding of Table 3.3 (the table is translated from the original Table 3.1). In the design of Figure 3.2, hue channel is used to code the two categories of data. Those cells that have been changed, and those that remained the same. In this design, “0” is coded as purple cell with a dot ("."), and “1” is coded as yellow colour cell with a letter to show the changed nucleotide in comparison with the origin viral sequence. The idea comes from IMAS [39] Multi-alignment view. I keep this design because hue channel is exceptionally powerful to encode categorical data [23].
THE COLOUR CODINGS IS CHECKED WITH VISCHECK WEBSITE\textsuperscript{1}, TO MAKE SURE THAT THE HUE COLOUR CHOICES FOR THIS ENCODING DESIGN WORKS FOR COLOUR-BLINDED PEOPLE. FIGURE 3.3(A) SHOWS DEUTERANOPHIE AS WELL AS PROTANOPHIE TEST RESULTS. DEUTERANOPHIE AND PROTANOPHIE ARE TWO DIFFERENT FORMS OF RED/GREEN COLOUR DEFICIT. FIGURE 3.3(B) IS TRITANOPHIE RUN THAT IS A BLUE/YELLOW DEFICIT. THIS COLOUR CHOICE WAS ALSO TESTED BY PRINTING THE DESIGN IN BLACK AND WHITE, AND IT WAS DISTINGUISHABLE.

\begin{figure}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
\textbf{Labels} & A & A & T & A & T \\
\hline
118 & . & C & . & . & C T \\
123 & . & C & . & . & C T \\
202 & G & C & . & . & . \\
211 & . & . & C & G & . \\
501 & . & . & . & . & C T \\
705 & G & . & . & . & . \\
\hline
\end{tabular}
\caption{The visual encoding of the first table}
\end{figure}

\begin{figure}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
\textbf{Labels} & A & A & T & A & T \\
\hline
118 & . & C & . & . & C T \\
123 & . & C & . & . & C T \\
202 & G & C & . & . & . \\
211 & . & . & C & G & . \\
501 & . & . & . & . & C T \\
705 & G & . & . & . & . \\
\hline
\end{tabular}
\caption{(a) deuteranope/protanope test result}
\end{figure}

\begin{figure}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|}
\hline
\textbf{Labels} & A & A & T & A & T \\
\hline
118 & . & C & . & . & C T \\
123 & . & C & . & . & C T \\
202 & G & C & . & . & . \\
211 & . & . & C & G & . \\
501 & . & . & . & . & C T \\
705 & G & . & . & . & . \\
\hline
\end{tabular}
\caption{(b) tritanope test result}
\end{figure}

\textsuperscript{1}http:
www.vischeck.com
3.2.3 Task Translation

The task translation step results in tasks abstraction in InfoVis words.

3.2.3.1 Task 1 Translation

Task 1 1.3.1 (Find columns that lead to the most severe rows.) of this study is translated into following steps:

1. Sort rows according to symptom severity
2. Find pertinent columns
   (a) Find non-pertinent columns and filter them out
   (b) Find pertinent columns (from smaller number of columns)
3. Relate interesting (pertinent) columns and group them together
4. Report the groups from most severe to least.

The last step is translated as a response to the study tasks (see Section 1.3.1) that ask the user to report a group of related columns for each danger level. Task 2 translation is very similar to Task 1, but instead of severe rows the analyst is looking to find the overall dangerous rows.

3.2.3.2 Task 3 Translation

Task 3 of this study (Between rows 123 and 51, which one is more similar to 583?), is translated in this section. This task is about pair-wise sequence comparison or evolutionary comparison. The two methods are translated in the following steps:

1. Look at the evolutionary tree and count the number of edges between 583 and 51 as well as 583 and 123.
   Or
2. Place the rows 583 and 51 beside each other.
3. Count the number of differences.
3. Repeat the same process with rows 583 and 123.

The following section describes the visual encoding of the translated tasks. Some of the steps of these tasks require adding interactions to the visual encoding of Table 3.3. Some others need to visualize Table 3.4 along with its interactions. Finally, some of the steps depend upon designing additional visual encoding and interactions.

3.2.4 Visual Encodings and Interaction Design

3.2.4.1 Sorting

The first step of the translated task (see Section 3.2.3.1) suggests enabling the user to sort the rows according to the values of different disease characteristics. The Matrix View consists of a table with sortable columns as well as a visualization of each characteristics levels. Figure 3.4 shows an early design of this view inspired by the Table Lens [11]. In this design, the length of each bar (horizontal bars ) represents the level number. Hue visual property is also used redundantly to code the different levels of characteristics in each column.

![Figure 3.4: The initial design of Matrix View employs bar lengths.](image)
Figure 3.5: Design of the Matrix View using position and colour saturation. Each column is divided by the number of characteristic levels for that specific column.

Sopan et al. [43] use colour saturation per cell for encoding different levels in their Table Lens visualization. Because the level of characteristics is an ordered data-type, according to Mackinlay’s heuristics list [23], the colour saturation should be a better alternative for the length channel for encoding this ordered information. Although using colour saturation seems a proper choice for encoding the levels, I use the position channel because it is the most powerful visual property for encoding all kinds of data [23]. Using the position channel, I change the initial design of the Matrix View. I also utilize colour saturations redundantly to encode the same property of the data.

In order to distinguish between different disease characteristics (distinguish between each column in Matrix View), the hue channel (different colours for each characteristics) is used. This is because the disease characteristics are nominal data and the hue channel is appropriate for separating different categories. Figure 3.5 shows the new visual encoding in Matrix View. Each column is divided by the number of its characteristics levels. The data ranges from the most dangerous number on the left to the least dangerous number on the right. Figure 3.6 illustrates the sorting interaction in the matrix. When a user selects a column in Matrix View, all the rows in both Matrix View and Main View are sorted all at once. This interaction is one way that Main View and Matrix View are linked to each
other.

As the translated Table 3.4 suggests, the user need to make a new aggregated column that consist of adding the existing columns. Figure 3.7 shows how the aggregated characteristics is created on exiting columns.

### 3.2.4.2 Filtering

In order to find the dangerous mutations for solving task 1 and 2 of this research study (see Section 1.3.1), the user needs to compare different columns with each other (addressing steps 2(a) of task 1 translation in Section 3.2.3.1).

As there are 1400 different columns in the encoded table, placing the columns close to each other frees up the cognitive load of the users and enables them to use their memory to focus on their desired task [40]. One way of putting columns close to each other is to allow the user to drag and drop the columns next to each other. However, enabling this feature admits that the user can change the natural order of nucleotides in a sequence. One must realize that the natural order is meaningful in the original domain. In order to keep the natural position orders, an alternative design decision is to enable the user to filter the columns that she/he is not interested in. By this filtering, the interesting positions in the DNA sequences are placed next to each other.
(a) The user wants to make a new column. (b) The new column is created by adding the existing columns. A new hue (blue) is associated to the newly created column to distinguish from other columns. The saturations of blue colour is used to show different levels.

Figure 3.7: How to make a new column from existing ones. Note that in order to let the user make ordinal values from numerical data, the system needs to map the different levels of the data to numbers. To make the design more effective, the user can choose the numerical levels associated with the ordinal levels. Also the distance between the levels can be varied. For example, “Severe” could be mapped to four, “Moderate” could be mapped to two, and “Mild” could be mapped to one.

While having this feature seems useful for exploring the data, finding relevant columns still requires manual work (exploring all the columns to find relevant ones). Therefore, in addition to letting the user filtering unwanted columns, an augmented filtering is designed to help the user exclude the unnecessary columns.

**Basic Filtering** Figure 3.8 shows how the user can separate out a group of columns (or one column). The appearing of hidden columns must be smooth and animated so that the view does not jump to a new state. The animated transition also enables the user to observe
CHAPTER 3. METHODOLOGY: DESIGN

the current process.

![Figure 3.8: Hiding the columns.](image)

**Augmented Filtering** A basic visual exploration of the visual encoding of Table 3.3 reveals that there are a number of columns with no substations in them. Since all the study involve the mutations, these columns do not contain any values for the solutions.

Furthermore, a small number of substitutions in a column may occur randomly and do not reveal any valuable information to the analyzers. Considering that some columns contain no information, a filter should be designed to eliminate the columns with different numbers of substitution. As Figure 3.9 explains, each level of the filtering deletes some of the columns with that number of substitutions. For example, filtering at level 0 hides the columns with 0 substations, level 1 further hides columns with 1 substations as well as columns with 0 substitution. The user can always go back to a previous level or to the unfiltered state.

**3.2.4.3 Finding**

As seen in Augmented Filtering, the system helps the user filter the irrelevant columns to work with fewer columns (less dimensions in the data-table). The second guide of the system in the filtering process is to help the analyzers find the columns that might be of interest to them for more investigations. (addressing step 2 (b)from the task 1 translation in Section 3.2.3.1)

**The P-Value View** There is a pattern within some of the columns that makes them interesting candidates regarding the Task 1 and 2. This pattern shown in Figure 3.10, suggests a relationship between substitutions in a particular column and the severity of the...
(a) The columns are not filtered.

(b) The columns with no substitutions are filtered.

(c) The columns with 1 or 0 substitutions are filtered.

Figure 3.9: Filtering the columns.
disease. In this figure, more number of substitutions occur in the most severe rows and fewer in the mild ones. This makes these positions and their substitutions to (hypothetically) be a candidate of causing severity of a disease. In general, assuming that all the rows are sorted according to the values of one of the disease characteristics from top to bottom, a significantly larger proportion of substitutions appear at the top rather than the bottom.

As humans do not complete pattern-detection tasks very well [28], we cannot rely on them to find this pattern in columns. In fact, it is very common that biologists have metrics to help them find interesting patterns. The Noblis team [4] uses the Mann-Whitney U tests p-value as a metric for guiding the user in finding relevant positions.

The Mann-Whitney U test states that the severe rows can be separated from others by splitting all the rows into two groups based on the existence of a substitutions in them. The null hypothesis is the lack of significant difference between the mean of the two groups. The
p-value of the tests is the probability of the null hypothesis being correct. The reverse of the P-Value (negative of the logarithm of the P-Value) suggests how much it is likely that we have a significant difference between the two groups. In this test, alpha is set to 0.5. If the reverse of the p-value is large, the null hypothesis cannot be rejected, suggesting that there is a significant difference between the two groups regarding each of the characteristics (for example, severity of the symptoms).

Representing the reverse of the p-value to the users guides them in finding relevant columns. The length channel is used, as the second most powerful channel after the position channel, for encoding the ordinal values of the p-value metric. Figure 3.11 shows the use of the length channel.

![Figure 3.11: P-Value visual encodings](image)

As the user might want to focus on those columns with the higher bar length, it is not efficient to merely hide/unhide all the other columns. Instead, it would be more productive to sort the columns based on the reverse of the p-value (length of bars). The problem is that in the original domain, the order of columns has a domain meaning and re-ordering
the positions is impossible. Considering this domain constraint, I suggest building another view that holds the p-value information and lets the user sort that columns in that view. There would be a link between the two views so that if the user’s mouse hovers on one column, the corresponding column in the other view is highlighted. I also suggest to keep the bars on top of the columns in Main View, so that the user could go over the bars while observing the columns’ pattern. To save space, there will be an option to turn off the bars in Main View. Figure 3.12 shows the sorted positions based on their p-value in P-value View. A mouseover on a column in P-value View will cause Main View to shift to contain that column. In Figure 3.12 this cause and effect action is shown by blue colour.

![Figure 3.12: P-Value mouseover effect](image)

There is also a slider-bar to filter the columns with the lower pattern occurrence at different levels. When the user filters columns in the second P-value View, the columns is filtered in Main View as well. Figure 3.13 shows one example of how the slider works in between the views. In this figure, the right P-Value is sorted the columns based on the bar lengths. However, after discussing the two separate views with domain experts, I figured out that the tool can let the users reorder the columns as long as there is an option to go
(a) There is a filter option to filter the columns based on the bar lengths.

(b) Some of the columns are filtered.

Figure 3.13: P-value filtering.
back to the original state. This suggests removing the second view and creating a button to sort the columns based on the p-value bar lengths. There is also a second button called “reset” that sorts the column according to their position number.

### 3.2.4.4 Grouping

In the translation of task 1 and 2 (see Section 3.2.3.1), users find the most dangerous columns first, then they find those columns that are related and group them together. Finally they rank the groups and report each of them according to level of danger.

The first and second steps of the translation of task 1 and 2, are already reflected in the design of the tool in “Sorting”, “Filtering”, and “Finding”. The third step of the translation of task 1 and 2 indicates that the user should find columns that are related.

The Basic Grouping section contains the details of enabling the user to make different groups. The user can separately load each group into the views in order to investigate the group information and to focus on the relationships between the columns. It is more likely that they will make these groups from the relevant columns. The design of Basic Grouping requires users to find the related columns manually. The Augmented Grouping design guides the users to find relationships between the columns so that the user can more effectively select the columns of the same group.

The grouping feature could also be used as a step in the translation of task 3. The user can choose to make a group of rows to focus on less rows (less data dimensions).

**Basic Grouping**  
Basic grouping feature is defined for both columns and rows. For separating the behaviour of the system, two modes is defined, row and column mode.

**Column Grouping**  
The idea is to let the user make different groups from a combination of different columns. The user can see an overview of the group and its general pattern. An overview of the distributions should allow analysts to see the big picture, and identify clusters, trends and outliers that may be candidates for detailed inspection [43]. Therefore, an overview of a group consists of a larger window than Main View information (prior to zooming).

To add a new group, the user clicks on the new group button (Figure 3.14(a)), followed by the renaming the new group (Figure 3.14(b)).

Figure 3.15 shows how the user can use the previously cerated group. Figure 3.15(a)
(a) User clicks on the New Group button.

(b) A textbox appears for group name.

(c) User clicks on the Add button.

(d) User selects a column.

(e) The selected column is added to the created group. An overview of the column is shown at the group.

Figure 3.14: Making a group of columns
contains some groups that are created by the user. An overview of each of the created
groups is shown at the Group View. If the user clicks on the overview (Figure 3.15(b)), the
content of that specific group is loaded in Main View 3.15(c). By clicking on the default
view (called “last”), Main View updates with the latest changes in the default group.

**Row Grouping** The row grouping is built to help users make groups of rows for
future analysis. All the features are similar to column grouping except that they user must
be allowed to click on rows and add them to a newly created group. When the user is in
row mode, the user can select different rows to make a group of them.

**Augmented Grouping** In order to guide the users to find related columns or related row,
an augmented grouping feature is designed. This feature is different for rows and column
because they could have different kind of relationships.

**Column Relations** There are two kinds of relationships between the columns that are
suggested in the VAST solutions. In this thesis, they are called Correlation and Complementary
patterns. Both relationships are defined between any pair of columns. Figure 3.16(a)
shows an example of the correlation pattern between a pair of column. In this figure, the
substitutions in both columns appear in the same rows. Figure 3.16(b) shows an example
of the complementary pattern between another two sample columns. The substitutions of
the two columns appear in the opposite rows meaning that wherever there is a substitution
in one column, there is no substitution in the other one.

Assuming we know the degree of the relationship between a pair of columns with numbers
from 0 to 1 for correlated ones and -1 to 0 for complementary ones, my initial suggestion to
encode this information was using a graph or a table representation.

Table 3.5 contains some results for an imaginary formula that codes the correlated and
complementary relationship for a set of columns. As this table indicates, the two columns
842 and 946 are highly correlated, whereas two columns 790 and 946 are complement to
each other.

In order to visualize the relationship between the columns, a separate view called Graph
View is designed for columns. As the name of this view indicates, the visualization consists
of a graph representation. Although the graph visualization is selected for the prototype
implementation, it is not the only options of representing this relationship. The following
section contains the explanation of this design as well as other alternatives.
(a) In addition to the default group which is called "last", two groups are created and named by the user.

(b) The user selects one of the groups.

(c) The content of the selected group is loaded in Main View (and all the other views). Selecting "last" group, the user can return back to original default group, which contains the entire data-set.

Figure 3.15: Example of utilizing the basic grouping feature to make different groups of columns.
Figure 3.16: Relationship between any pair of columns.

Table 3.5: Complementary Relationship

<table>
<thead>
<tr>
<th></th>
<th>946</th>
<th>842</th>
<th>223</th>
<th>790</th>
<th>269</th>
<th>821</th>
</tr>
</thead>
<tbody>
<tr>
<td>946</td>
<td>1</td>
<td>0.9</td>
<td>0</td>
<td>-0.8</td>
<td>0.6</td>
<td>0</td>
</tr>
<tr>
<td>842</td>
<td>0.9</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>223</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0.7</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>790</td>
<td>-0.8</td>
<td>0</td>
<td>0.7</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>269</td>
<td>0.6</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>821</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
**Graph View** The first option to show the relationship between a pair of columns is using a matrix visualization [15]. Figure 3.17 shows the example of this view for the table 3.5. Each cell is represented by a row and a column in the matrix. An entry in the matrix is coloured by different saturations of blue and red. Blue colour codes the positive (+) values and red codes the negative (-) values in correlation results. One benefit of using this matrix is, by re-arranging the rows and columns, some interesting patterns between a group of columns would be revealed. However, this option requires a large screen space (at least 1400 × 1400 pixels). Even in the case of cutting the first half of the matrix (because it is symmetric), it still remains very large for a normal computer screen space. One drawback is that we cannot eliminate the cells with 0 correlation from the space.

The second option is using a node-link graph, where there is a link between a pair of columns only if their correlation is non-zero. The link is coloured blue for correlation (numbers greater than 0) and red for complementary (numbers less than 0). Colour saturations and line weights are also redundantly used to encode the same information.

As there are a considerable number of columns with zero correlations, this option conserves the space better than the table representation. In Figure 3.18 each node size also corresponds to P-value View bars. The reason is that the user wants to relate the interesting positions. The node size remains for the paper prototype and it is not reflected in the software prototype.

This view is also linked to Main View, so when the user filters the columns with different number of substitutions or by the p-value measure, the corresponding nodes will be filtered.
There is also another filter mechanism built into the view that removes the different levels of correlation links. Figure 3.19(a) shows the original position of the two filters. Figure 3.19(b) shows how moving the sliders removes some of the corresponding links.

**Column Relation Metric** The initial idea for showing these two relationships was representing the result of "Pearson’s correlation." calculation for any pair of columns. Equation 3.1 defines these relationship with the two columns X and Y using "Pearson’s correlation." The two sets of data are defined as two sets of data contains by 0 (for purple cells) and 1 (for yellow cells). In this equation, the covariance of the two variables divided by the product of their standard deviations. The E operator is the expected value that is defined in terms of means of each of the two data columns.

\[
corr(X, Y) = \frac{cov(X, Y)}{\sigma_x \sigma_y} = \frac{E[(X - \mu_x)(Y - \mu_y)]}{\sigma_x \sigma_y}
\]  \hspace{1cm} (3.1)

The Pearson correlation for any two sets of data is ranged between -1 to +1. It takes its
extremums when one set is a linear combination of the other one. The closer the number is to 1 (or -1), the stronger the correlation (or complementary) is between the two data sets. Values close to zero, indicate lack of linear dependency.

Given that the cell substitutions in this data-set are represented by values 0 and 1, one may define the similarity between two columns with measures such as Pearson correlation. This is however not optimal due to many zeros (no substitution) occur in the columns, which results in correlation close to 1.

To alleviate this problem, I propose to use a new measure which ignores the common
zeros between columns. Assuming that two columns, \( X \) and \( Y \), each have \( n \) members, 
\[ X = \{x_1, \ldots, x_n\} \] and 
\[ Y = \{y_1, \ldots, y_n\}. \] The measure is defined as follows:

\[
M(X, Y) = \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \oplus y_i, \tag{3.2}
\]

where \( \oplus \) is the logical XOR operation and results in 1 when one of the side equals to 1 the other side equals to 0. This measure, ignores entries with no substitution in both columns, increases when entries with substitutions occurs together and decreases when substitution complements each other. Given that, both positive and negative values are expected.

**Row Relation** The relations between rows are hierarchical. The already designed Graph View is used to make a Tree for the representation of this relationship.

### 3.3 Implementation

The four primary languages for implementing visualization concepts are prefuse \cite{16}, Protovis \cite{2}, D3 \cite{3}, and Processing \cite{13}. FilooT is implemented in Processing programming language\(^1\). Processing is an open source programming language and it was helpful to sketch software ideas and make a quick working prototype. The code structure follows the MulteeSum open-source project \cite{25}.

In order to get feedbacks from the domain experts, I quickly developed a prototype from the design sketches. The designs can be changed dramatically as a result of the user study. Therefore, I did not focus on low-level usability testing on FilooT. In fact, the advantage of a prototype, as opposed to a fully developed system, is that the programmer can reflect the changes more easier. In the current implementation, due to time limitation, some design options remain on-paper prototypes.

In fact, prior to make a new tool (FilooT), I intended to implement my ideas within IMAS framework. To learn IMAS code, I started documenting its source code which includes visual models of the source codes using the Unified Modeling Language (UML). Because IMAS was an already developed tool, the UMLs were created in a reverse engineering process. Although these diagrams could communicate to future developers to make changes and maintain the software, creating them shows one of the research pitfalls that I went in

\(^1\)http://processing.org/
to. This drawback is called the non-rapid prototyping pitfalls \cite{38}. While at the time I went into that drawback and spent a lot of time on documenting the system instead of focusing on the design, nevertheless the diagrams could be useful when a future researcher wants to implement the final designs of FilooT in IMAS that is a more general framework.

### 3.3.1 Screenshots of FilooT

To explore the VAST Challenge 2010, Mini Challenge 3 data-set, I designed and implemented FilooT, a Processing prototype for visual data analysis. This section shows the images of FilooT.

#### 3.3.1.1 The Overall System

The primary interface for FilooT is shown in Figure 3.20. Main View is the purple box in the middle. Matrix View is on its left. The left panel is Group View, containing overviews of the groups that the user created. There are two modes of the system: Row and Column. The right panel provides Graph View, displaying nodes and links associated with the current visualization mode. In addition, for each column, the reverse of the p-value metric’s visualization is available in P-Value View, which is located above Main View. During the analysis process, the Status-bar informs the user that their selected action is done or is in progress.

On the left bar, there is Group View and some groups that the user created before is shown. If the user is in Row mode, he/she can select rows to make a group and if they are in Column mode, they can select columns. The left upper buttons are to select the mode. The left down side is the status bar that shows the action that the user took. For example, the user ”added the column 400 to a new group”.

#### 3.3.1.2 Mode Changing

Figure 3.21 shows the mode changing buttons. By selecting each mode, the following features change:

- When they move their mouse in Main View, the row labels (sequence headers) are highlighted to show which row (sequence) is being explored. On the other hand, if the user selects column mode, position headers are highlighted.
Figure 3.20: FilooT visualization system. (a) An interactive visualization table to represent the genetic sequence information. (b) A matrix visualization for interacting with the disease characteristics data. (c) The P-Value bars to show a metric (reverse of pvalue in Mann-Whitney U test) about each column. (d) The Group View containing the user created groups along with an overview of each group. (e) A graph visualization for representing row (or column) relationships depending on the system mode (Row based or Column based). (f) Two buttons enable the user to choose between the Column and Row mode. (g) The Statusbar is being updated after each action that the user makes.

- The graph view updates based on the mode. When the Row mode is activated, the nodes are the rows and the edges show the relationship between the rows. If the user selects column mode, the nodes are columns and the edges show the relationship between the columns.

- When the Row mode is activated, the user can make groups of rows. The user can add rows to a newly created group by selecting a new group and clicking on rows in the Main View.

- When the Row mode is activated, a row of the overview of the selected group will be highlighted depending on where the mouse is in Main View (Brushing and Linking).
3.3.1.3 Main View

Figure 3.22 shows Main View that is an interactive tabular visualization for the genetic information about each viral sequence. The first row shows the genetic information about the original viral sequence. The second row shows position numbers. The numbers start from one and end with the length of the sequences (1404 for this data-set).

Each row indicates one sequence. Each cell information indicates the result of the comparison of each sequence with the original sequence appeared in the first row. The purple cells are those that did not change in comparison with the original sequence. The yellow cells represent a change in a particular row and column. The letter indicates a change in the information of the specific cell in comparison to the corresponding column in the original sequence.

In figure 3.22, Row mode is activated, and the sequence 175 header is highlighted to show the mouse position.

As seen in Figure 3.22, the position number in the second row does not include all the numbers between any two position numbers that are beside one another. This observation indicates that the user might have used a filter (from Main View, P-value, or Graph View).

Moreover, the numbers in the second row are not in sequential order. That change in order of the column indicates that the user might have sorted the columns according to the values of P-value View. In that case, the user can go back to the original ordering by selecting “reset” button in Main View.

The scroll bars at the bottom and on the right are so that the user can explore more of the sequences and the positions’ data.

The “+” and “-” buttons allow user to zoom in and out. This zooming will also affect Matrix View and the rows in Matrix View will be zoomed as well (see Section 3.3.1.4).

The “reset” button returns the columns to their original sequence from one to the length
of the sequences (1404 for this data-set). This feature is used whenever the user previously changed the column positions, and want to reset the position numbers.

The Main View’s filter exclude the columns that have fewer yellow cells than the filter number. This filtering affects the other views linked to Main View. More specifically, the corresponding columns in Graph View and P-Value View are filtered.

3.3.1.4 Matrix View

Figure 3.23 shows the Matrix View beside the Main View. The two views are linked together in three ways:

- When the rows’ positions are changed in one view (for example if the user sorts the rows), their vertical positions will be changed in the other view. The two views share the row labels.
• The zooming will affect both views. Figure 3.24 (a) shows a fully zoomed out Matrix View. The user utilizes Main View’s “+” and “-” buttons to zoom into both Matrix View and Main View.

• At the bottom of each column in Matrix View, an overview of that specific column is provided so that the user can see the pattern of the change for all the row values for that specific disease characteristics column, without the need to zoom. When the Row mode is activated, and a sequence header is highlighted to show the mouse position, it also highlights a row in the overview of Matrix View. Figure 3.23 shows a row that is highlighted in the overview.

Figure 3.23: Matrix View and Main View are linked together.

Below is the list of interactions that the user can perform in Matrix View:
(a) A fully zoomed Matrix View.

(b) Ascending sorting. The higher values are on top.

(c) Descending sorting. The higher values are at the bottom.

Figure 3.24: Matrix View.
• In Matrix View, if the user selects a column header, the rows (in both Matrix View and Main View) will be sorted according to the values of that column. On top of each column, there is a coloured label that shows the different levels in that particular column. The darker the colour, the more severe the characteristics of the disease. The coloured labels are placed by severity from right to left.

• The user can choose between ascending or descending sorting. Figure 3.24 (b) shows when the user selects to sort the rows from high to low. Figure 3.24 (c) shows when the user selects to sort the rows from low to high.

• The “add” button, (potentially) enables the user to make a new column by combining the existing ones with a simple mathematic function in between them (such as plus). The user also can multiply a column by a number. This feature remains mockup and by the time of the study was not implemented. Although I explained its potential behaviour to the user.

• The user can zoom in and out to the view using “+” and “-” buttons from Main View.

3.3.1.5 P-value View

Figure 3.25 shows P-value View. The bar length is the reverse of p-value in the Mann-Whitney U test. The filtering feature of this view enables the user to filter out the columns that their bar lengths are smaller than the number on the filter. This view also lets the user sort the positions based on the bar length. The columns will be sorted from high to low and placed from right to left. Figure 3.25(a) shows P-value View before the user sorts or filters the columns. Figure 3.25(b) shows the effect of filtering the columns and sorting those columns based on their P-Value bars.

3.3.1.6 Group View

Figure 3.26 shows a group that the user created while Column mode is activated. An overview of this group’s genetic information is shown in Figure 3.26(a). Figure 3.26(b) is using to show the linkage between Main View and the selected group. When the user selects a group among the previously created groups from Group View, the chosen group’s data will be uploaded into the system. Therefore, the data in Main View matches the data in...
CHAPTER 3. METHODOLOGY: DESIGN

(a) A column label and its corresponding bar is highlighted to represent the mouse position on a column.

(b) The columns are sorted from left to right. The user also filtered some columns with smaller bars.

Figure 3.25: P-value View

the selected group. As figure 3.26(b) illustrates, when the user’s mouse hovers a column, the corresponding column is highlighted in the overview of the currently selected group.

There is a predefined group in each Row and Column mode that contains the entire data-set for the user to be able to go back to the original step.

3.3.1.7 Graph View

Figure 3.27’s Graph View shows the relationship between the rows in Row mode. In this figure, the user’s mouse position highlights the corresponding row label in Main View (in Figure 3.27(a)), as well as the equivalent node label in Graph View (see Figure 3.27(b)). If the user deletes a row from Main View, the matching node in Graph View will be deleted.

When the user is in Column mode, the graph nodes are columns and the links are their
relationship. Two filters are placed to enable the user to sort out the columns based on the strengths of their connection. Figure 3.28 shows the two filters.

3.3.1.8 Status-bar

Figure 3.29 displays the last user’s action. This Status-bar gives feedbacks to the user’s actions. For example, if the user clicks on a row while the new group is selected, the consequence of this particular clicking is that the row is added to a new group.

3.4 Usage Scenarios

The following scenarios are my own use of the tool to solve task 1 1.3.1 and task 3 1.3.1 of the study. Task 2 1.3.1 is not included in this section because it very similar to task 2.

3.4.1 A Usage Scenario for Solving Task 1

Task 1.3.1 (Identify mutations that lead to an increase in symptom severity) was solved in the following manner:

1. Select Column mode.
(a) A row is highlighted in MainView to illustrate the user's mouse position on the rows.

(b) The matching node from Main View to Graph View is highlighted.

Figure 3.27: Graph View in relation to Main View.

Figure 3.28: Graph View has two filters to delete the columns in Column mode from the column graph.

Figure 3.29: Statusbar: the user has added row “51” to a newly created group while in Row mode.
2. Sort the rows by clicking on the Severity column in Matrix View.

3. Filter the columns with fewer yellow cells (mutations) than the number shown in Figure 3.30’s slider-bar.

![P-value View](image)

Figure 3.30: Task 1 Main View and P-value View.
4. Use P-Value filter and delete the columns with shortest bar. Four columns now remained to investigate.

5. As Figure 3.31 and Figure 3.30 show, column 842 (highlighted) and 946 have similar pattern. The other remaining columns, 161 and 790, also have similar patterns, and they could be complements to the first two. Figure 3.32 in Graph View that confirms this hypothesis by showing the blue and red links between the columns. Graph View is helpful for understanding the relationship between columns.

6. Sort the columns according to their P-value bars.

7. The first answer to task 1: 946, 842, 161.

3.4.2 A Usage Scenario for Solving Task 3

The following steps were used to solve Task 3 1.3.1: (Nicolai has a strain identified by sequence 583. One patient has a strain identified by sequence 123 and the other has a strain identified by sequence 51. Which patient contracted the illness from Nicolai and why?)

1. Select Row mode.

2. Create a new group while in Row mode. Then, select the relevant rows which are 51,123 and 583.
3. Select the new group. Thus, the entire data-set is replaced with the information about these three rows. Figure 3.33 shows that all the rows in Graph View are have been removed. As tree structure demonstrates in Figure 3.33(b), row 123 is closer to row 51, which indicates that row 123 contracted the illness from Nicolai.

4. Figure 3.34 provides an alternative solution to Task 3, using Main View 3.33(a). Assume the row which displays the most similarities to row 583 belongs to the patient got the disease directly from Nicolai. In figure 3.34(b), the columns are filtered using the Main View slider. The columns with no yellow cells are deleted. Figure 3.34(a) contains Main View overview. As these two pictures evidence, there are fewer differences between rows 583 and 123 than between rows 583 and 51.
a) Main View holds the three rows: row 51, row 123 and row 583.

b) Graph View nodes are the matching rows. Although the other nodes are deleted, the structure of the relationship between the rows remains in Graph View.

Figure 3.33: In order to solve Task 3 of the study, while in Row mode, the user created a new group consisting of three rows.
Figure 3.34: The linkage between Main View and Group View in Task 3 Solution.
Chapter 4

Methodology: Evaluation

Choosing an evaluation methodology for Information Visualization tools is a challenge for the researchers. One of the main reasons of this difficulty is that this field has diverse intersections with many other fields such as psychology, semiotics, graphic design, and art [51]. However, there are a number of published InfoVis papers that dedicate at least one section to the evaluation methodology. Each papers has an example of applying different evaluation approaches to an InfoVis system. Moreover, there are some surveys and framework papers that aim to guide researchers to choose their evaluation methodology among the existing approaches that fits their research question.

4.1 Evaluation Methods

4.1.1 Design Stage

I have looked to my research from a high-level perspective among available InfoVis methodologies. The framework that I choose is called the Nested Process Model for Visualization Design and has four design stages: domain problem characterization, data/operation abstraction, encoding/interaction technique design, and algorithm design [27]. This model has guided my search by providing a taxonomy of the existing stages of the work in this field and some evaluation methods for each stage (layer). The design nature of my work falls into level two and three of the Nested Layer Framework [27]. In Figure 4.1, the four layer model is re-drawn for this thesis study to show each layer’s contribution in my thesis.

As [38, 27] suggest, one single study usually do not contain all the different aspects.
Specifically in this research, I have used a benchmark data-set/task-set and so I am assuming that the tasks are already validated and they reflect the target domain users work. I also did not design any efficient algorithms for my visualizations. Therefore, the algorithm layer and domain problem characterization layer from the Nested Framework [27], are excluded in this research.

The following is the two important recommendations of the Nested Model to avoid the threats at the visual encoding and interaction level layer. Under each guide, I have shown how my chosen evaluation methodology followed this advise.

![Diagram](image)

**Figure 4.1**: The Nested Layer framework for this study. Redrawn after Figure 1 of [27]

**The design** needs to follow perceptual and cognitive principles [27]. I have used a number of heuristics for information visualization [51] to follow this recommendation. In chapter 3, there is an extensive discussion of design choices with immediate justifications using design guidelines as it is recommended in the nested model. The discussions was done on the early sketch prototypes.

**The design** should be able to communicate with the analyzer and be useful towards their problem solving [27]. The goal of such InfoVis tools is to support the target domain users in their tasks. In the case of this study, I need to assess the initial design idea with domain experts to see to what extend the tool supports Bioinformatics users to solve the study’s tasks.

My research inquiry as part of the design process was achieving a richer understating of strengths and weaknesses of the design to make it better iteratively. My goal with respect
to this recommendation was set to know whether my design is effective enough for solving the tasks problems.

4.1.2 The User Experience Scenario

As from the past section, I was looking to critique my design choices with the help of an evaluation methodology, to be able to iteratively improve the different aspects of the design (or suggest and improvement as a future work direction).

In order to decide about an effective evaluation for my research, I mapped my research into a specific scenario from the seven guiding scenarios for InfoVis evaluation [21]. The seven evaluation scenarios are as follows: evaluating visual data analysis and reasoning, evaluating user performance, evaluating user experience, evaluating environments and work practices, evaluating communication through visualization, automated evaluation of visualizations, and evaluating collaborative data analysis. Each scenario is defined based on a link between evaluation goals and evaluation approaches [21]. Yet, the first recommended step towards selecting a methodology (or a scenario), is to specifying a clear goal for the evaluation [8].

So I defined my general research question as: "How the design of FilooT could help the domain users solve the tasks problems". As the nature of this research question suggests, I conducted a qualitative study at this level of the design. This selection requires an implemented system so that the users could interact with the visual encodings to solve the tasks problems. For that purpose, I implemented the early sketches with Processing language\(^1\) and that was how FilooT was born. The name of this tool comes from the several filtering option that this tool provides for the user.

As Ellis et al. [10] suggest, my research falls into the Formative category because the purpose of the study is to inform the design rather than summarizing the effectiveness of the tool (Summative Research [10]). Considering the formative nature of my research question, I identified my work as User Experience scenario(UE) from the seven guiding scenarios for InfoVis evaluation [21]. The User Experience scenario is to look how the participants react to the tool with the goal of understanding how much the tool is helping them solving some specific tasks with the final goal of informing the design [21]. The next step for evaluation methodology was to address the main questions in UE category which is “what do my target users think of the visualization?” [21].

\(^1\)www.Processing.org
To answer this question, I came up with a range of specific questions that their answers could help me reaching towards understanding the tools from the users point of view. The evaluation approach of asking these questions comes from combining three of the four suggested approaches for UE evaluation, which are Informal Evaluation, Usability Test and Laboratory Questionnaire \[21\]. One limitation of the selected method is asking the users what they think is accurate. Usability inspection methods such as the Cognitive Walkthrough method \[41\] is an alternative for resolving that specific issue. However, the evaluation methodology that is used in this thesis is a user-based method in which problems are found through the observation of and interaction with users while they use or comment on the interfaces.

4.2 Study

4.2.1 Method

This study is qualitative and it falls in to User Experience category \[21\] for system evaluations. This includes presenting the system to the user; let them play \[21\] with it to answer some tasks with the goal of understanding the design flaws and potential usefulness. The study follows by an informal interview with the user to understand their opinion about the tool. The main research question to be addressed is what do my target users think of the visualization? The open-ended questions of the interview are designed to address this question.

Although for this study, I had a set of pre-defined tasks (see Section 1.3.1) which leaned towards the Usability Testing category, I explained the users that the process of solving the tasks is more important than reaching to answers. This approach was about Informal Evaluation category. So at the beginning of the process, I briefly mentioned the tasks and the dataset which was a little towards to an informal usability testing.

Also, my approach leans a little bit towards the Informal Evaluation as I asked the participants to freely talk about anything comes to their mind about the tool during and after the process.

I recorded field-notes from my observations of user-interactions of the tool, and their expressed ideas/feelings.

The followings are the interview guide questions.
• How each view/features helped you to find the answers? Any of them were more or less useful?

• Did you have any difficulties understanding any of the presented information in any of the views?

• Please explain your process and steps of finding the answers.

• Please give me your feedbacks to make each of the views better. Your suggestions may include your ideas of new interactions or refinement of some parts of the design. [Same line of questioning for each of the views]

• Are there limitations of the current system that would hinder its adoption?

4.2.2 Settings

A laptop computer was used. The studies took place in lab environment.

4.2.3 Participants

The study participants are live subjects in the lower mainland. The subjects are undergraduate/graduate students or postdoctoral researchers. All the participants are over 19 years old. The requirement for the study is that the participants need to be familiar with DNA multiple sequence alignment concepts in bioinformatics, and they should be interested to participate in this study.

4.2.4 Process

The experiment took less than 1.5 hours.

1. They were asked to read and sign the consent form

2. They were asked to fill out a questionnaire (PRE-STUDY QUESTIONNAIRE) Appendix C with a few questions about their familiarity with the domain and their experience with similar tools.

3. The Study Task/Data Description (STUDY TASK/DATA DESCRIPTION) Appendix C was given to them. After they read it, they were trained for 10 minutes to learn to use the basic features of the software.
4. They used the system for 30 minutes. They were encouraged to write down their findings, think aloud and express their thoughts, concerns and their questions at any time during the study. I used pen and paper and took notes from what I observed from their use of the tool. It is worth mentioning that one limitation of this study is that the given time can interfere with their solving task. However, the accuracy of the answers are not studies in this research, and only the process of the problem solving using the tool is considered.

5. After the 30 minutes passed, we had a semi-structured interview about their experience with the tool. There were open-ended questions.

6. I thanked them for their participation in this study

7. The received the compensation for participation, and signed the compensation form (COMPENSATION RECORD) Appendix C.

4.2.5 Ethical Considerations

With regard to research ethics for research involving human participants specifically, no information was withheld from participants.
Chapter 5

Results and Discussion

This chapter presents and discusses the study findings. The results of the user study led me to create a list of requirements as well as develop a guideline for future work. These results informed the initial design of the tool [6].

5.1 Study Results

Table 5.1: The organization of the users’ ideas

<table>
<thead>
<tr>
<th>View</th>
<th>Visualization Comments</th>
<th>Interaction Comments</th>
<th>Observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main View</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matrix View</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P-value View</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Graph View</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Group View</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Row/Column</td>
<td></td>
<td>General Comments</td>
<td></td>
</tr>
<tr>
<td>User General</td>
<td></td>
<td>Comments and Suggestions</td>
<td></td>
</tr>
</tbody>
</table>
During the study session, the participants used the tool to find the answers for three tasks (see Section 1.3.1). I used paper and pen to write their comments about the system features. They were also asked to share their findings about the system. There was an interview with open-ended questions to capture their comments at the completion.

In Table 5.1, I classified the comments on the various views and features of the system according to six categories: Main View, Matrix View, P-value View, Graph View, Group View, and Row/Column modes.

5.1.1 Main View

5.1.1.1 Main View Comments

The list below contains the comments on Main View. Main View (see Section 3.3.1.3) is the only view FilooT and IMAS have in common, and it adds more interactions to IMAS multi-alignment view.

Visualization Comments

- The colour choice for mouse hovers (changing from gray to pink) is not discernible for individuals with colour-blindness.
- One user suggested we choose different colours for each nucleotide.
- The same user suggested that we choose similar colours for T and A, and similar colours for C and G, but that the two categories should be differentiated.
- The labels must be readable, regardless of the zoom level.

Interaction Comments

- According to users suggestions, the sliderbar was the most usable feature.
- Another user asked if it would be possible to sort the rows on their ID, and to search for rows by typing their labels directly into a textbox.
- If the fonts is small the labels could pop up in response to a mouseover.
- One user wanted to be able to type a motif and have the system highlight it with its local alignment score. This would be useful in order to know if a column is in a
CHAPTER 5. RESULTS AND DISCUSSION

conserved region or not (conserved sequences are similar sequences that occur within DNA sequences). Another user mentioned that the system could be linked to the UCSC Genome Browser \(^1\) to coordinate data so as to be able to highlight information about adjacent columns.

5.1.1.2 Main View Observations

- The sliderbar was the first and most common interactions with the tool.

5.1.1.3 Main View Discussion

- Regarding different substations in a column, the user pointed out that in a real data-set different kind of substitutions usually happen in a column, yet there is only one kind of substitution in a column in this study’s data-set. For example, if an ”A” changes to ”C” in one row in one column, then all other substitutions in that particular column will also be a ”C”. By contrast, in a real data-sets, there are the four kinds of substitutions in a column. To make the tool adaptable to a new data-set, as some of the users already suggested, I could use different hues to show different nucleotides (Figure 5.1).

One potential drawback for this suggestion is that it increases the number of hues in the tool. Depending on the number of columns in Matrix View, this could interfere with the hue uses in that view, because using more than eight hue colours on screen is not recommended (see Section 3.1.1). Instead, I suggest using only one hue per column for Matrix View and distinguishing between columns by adding extra space between them. Figure 5.2 shows this idea. More specifically I suggest using Proximity to show the organization of inter and intra columns (see Section 3.1.1 ). Also the designer must prevent hue overloading. For instance if they want to use red for showing one of the nucleotides, they should use a different hue for edge colours in Graph View.

- The data-set carries no information about adjacent columns. According to the participants’ comments, authentic data-sets contain these information. Figure 5.3 shows an example of neighbouring columns which do have some relationship with each other. To solve the tasks, researchers need to know if a column is related to its neighbours.

\(^1\)http://genome.ucsc.edu/
CHAPTER 5. RESULTS AND DISCUSSION

Figure 5.1: Users suggestions led to a change the number of colours used in Main View.

and what the nature of that relationship is. Examples of these relationships include codon and motif information in data-sets. However, because the VAST Challenge stated that the DNA is non-coding, codon analysis and AA sequence analysis cannot be considered.

- All of the users were familiar with at least one Visual Analytics tool similar to Main View. Although some of the interactions that they used to see in similar tools were not relevant to the study tasks, the users desired to see all of the familiar features in
CHAPTER 5. RESULTS AND DISCUSSION

Figure 5.2: Increasing the number of hue in Main View led to using limited number of hue in Matrix View. Instead of Hue, Proximity is used to differentiate columns (see Section 3.1.1).

Figure 5.3: neighbouring columns contain some information in many real-world data-sets.
5.1.2 Matrix View

5.1.2.1 Matrix View Comments

Below are users’s opinions about Matrix View. Matrix View (see Section 3.3.1.4) has an overview to represent the overall trend on each column. A row of this overview will be highlighted when the user is in Row mode and the user’s mouse hovers over that particular row.

Visualization Comments

- A user thought that the as yet unimplemented “add” button could be extremely helpful.

Interaction Comments

- A suggestion was to have the system show the row label in response to a mouseover.

- When the user clicks on the coloured label at the top of each column, all the rows will be sorted according to the values of that column’s characteristics 3.3.1.4. However, when the user clicks again, the rows order might be changed because the sorting algorithm allows for sequences that are different but still correct. Therefore, although the rows will be sorted each time a user clicks on a coloured label, the rows’ order within a level might be changed. The user did not like this and preferred consistency and predictability in the column sorting behaviour.

- Another user liked that the coloured labels at the top of each column were clickable and the clicking on them would sort the rows (see Section 3.24). (He said, “It’s pretty cool”. Moreover, he thought that these labels could be made more useful if the different levels were clickable separately so that the system would jump to a state in which Main View and Matrix View contained the rows with the selected level.

- One user suggested that Matrix View could have a built-in option to keep track, and sort the next column based on previous selections. He mentioned that could be particularly useful if the analyzer had certain priorities of columns and want to see a
level of disease characteristic in that priority list. The user opened a Microsoft Excel spreadsheet and showed me that he uses this feature on a daily basis.

5.1.2.2 Matrix View Observations

- One user sorted the rows based on different characteristics and used the Overview and P-value bars to see which characteristics contributed more to overall danger.

5.1.2.3 Matrix View Discussion

The compilation of the user’s comments on Matrix View shows that this is one of the most frequently used views. The users’ suggestions could be used as a guideline to make this view more accessible and useful for completing the study tasks. Below is my extension of how one of the suggestions should be considered for further implementation:

- For the user comment about clicking the coloured label in order to update the view with new information, I suggest that instead of Main View suddenly switching to new information, the system automatically move the vertical scrollbar which shifts Main View until the new, selected column is reached. Through this process, the user sees that the change in Main View is fluid, while keeping track of how the information changes.

- The above strategy could be used to link Graph View and Main View. When the user clicks on a node in Graph View, Main View needs to contain the matching row. In the case of a row needing to appear in Main View window, vertical scroll bars could be used. In the case of a column, horizontal scroll bars could be used to move Main View columns.

- It seems implementing the "add" button will be useful. Levels need to be associated with a particular number, and this number should be editable by the users.

5.1.3 P-value View

5.1.3.1 P-value View Comments

P-value View elicited a few comments. This view has a section that filters out the columns based on the reverse of the p-value of the Mann-Whitney U statistical test that suggest a pattern in the columns (see Section 3.2.4.3).
CHAPTER 5. RESULTS AND DISCUSSION

Visualization Comments

- In lieu of the p-values, the text should show its logarithms, which is used for generating the bars’ length. The p-value itself is not useful.

- In the current version of the tool, P-value bars are calculated only for overall disease characteristics. One user said that if the bars were updated for each of the disease characteristics, he would employ this view for solving task 1 in addition to having used it for task 2.

P-value View Interaction Comments

- One user mentioned that he will always filter the columns with the least bar lengths.

5.1.3.2 P-value View Observations

- P-value sliderbar is used to filter the columns that have mutations in all the rows. Main View Slider-bar could be used to filter those columns as well.

- The users immediately filtered out the P-value =1 columns for task 1 and 2.

5.1.3.3 P-value View Discussion

- One user had a very interesting observation about the slider-bar in Main View while using the slider-bar of P-Value view. He pointed out that there are some columns in Main View table containing a lot of mutations or yellow cells. If a column consists of all mutations, either there is a mistake in original native sequence, or the slide-bar of Main View should be able to filter it. If there is indeed a mistake, this observation shows how InfoVis could be used to understand the reliability of the data [26]. The reason why filtering of those columns will be useful is that they contain no information. They are very similar to the columns with no mutation.

- P-value should be calculated for each disease characteristics. When the user selects a column from Matrix View, the corresponding information should be updated in P-Value View.
5.1.4 Graph View

5.1.4.1 Graph View Comments

Participant had various opinions about Graph View (see Section 3.2.4.4). Graph View representation has two different modes. One to shows the evolutionary relationship between the rows (see Section 3.2.4.4), and the other to shows relationship between the columns (see Section 3.2.4.4).

Visualization Comments

- One user thought that the measure for calculating edge weights for Column Mode Graph might not be correct. He suggested the Mutual Information measure could be used for that purpose.
- One user said this view was useful for tasks 1 and 2, but not as much as P-value View.
- The tree structure could show the row clusters better if the node position changed so that each node correlated to its matching row in Main View.

Interaction Comments

- A user wanted to be able to click on a node in Graph view and add it to a new group. Currently this is only possible in Main View.
- Another user wished he could select multiple nodes by holding down the Shift key in Graph View and see the matching information be highlighted in all the other views.
- When one user clicked on a node in Graph-view, he expected Main View to shift to contain the matching row (or column).
- If a user deleted some columns, he wanted Graph View for Row mode, which is a tree representation, to be updated to show the new structure between the rows.

5.1.4.2 Graph View Observations

- One user applied the slider-bar to filter the nodes with correlation numbers between 0 and 0.5.
CHAPTER 5. RESULTS AND DISCUSSION

5.1.4.3 Graph View Discussion

- In general, it seems that the domain users do not find that the metric used for calculating the column relationships (see Section 3.2.4.4) is reliable. The users mentioned that a metric must have already existed in Bioinformatics to calculate the edge values precisely. One user suggested that the Mutual Information measure might be a better metric. From this comment, I concluded that one of the limitations of this study is that the metrics were calculated by the information available in previous VAST Challenge submissions. Confirming the metrics’ validity, could have been done before creating the visualization.

- Another interesting suggestion is to change the Tree layout for rows (in Graph View in Row mode). Each node should be correlate to its matching row in Main View. Having such layout would enable the user to sort the rows based on clusters that appear in the tree structure.

- When the user selects a node in Graph View, she/he expect to see the corresponding column/row in Main View. Main View could shift itself to contain the selected row/column. This shift could be implemented smoothly using animated transitions. The scrollbars would also move while Main View was shifted by the system (see Section 5.1.2.3).

5.1.5 Group View

5.1.5.1 Group View Comments

Participants’s also had comments on Group View (see Section 3.2.4.4). This view allows the users to make groups of column in column mode and groups of rows when the system is in Row mode. These groups are not visible to the user in the other mode; rather, they are only visible in the same mode in which the user created them.

Visualization Comments

- For one user, he said that the overview clearly showed the pattern and it is interesting.

- In the current version of the tool, the scrollbar in Main View moves Group View’s overviews. The overview should contain all the rows so that the vertical scrollbars do not have impact them.
CHAPTER 5. RESULTS AND DISCUSSION

Interaction Comments

- For one user, he created a new group of rows because he found that he could not filter the un-wanted rows.

- Out of four users, only one utilized the creating new group feature for creating a group of columns.

5.1.5.2 Group View Observations

- One user preferred to look at the pattern with Main View’s zooming feature in Main View rather than using the overviews of the groups. It seems that they want to create the overview themselves.

- One user said the overview is very interesting. Nevertheless, during the process he used Main View’s zoom feature rather than the overview in order to see pattern.

5.1.5.3 Group View Discussions

- It seems that unless the overview of each group reflects all the information in the created group, it is not useful. In the current version of the system, the overview of each group shows a larger window (in comparison with when the sequences of the group information are loaded and shown in Main View), and not the entire data. This limitation could be one of the reasons for not using the overview feature, and instead, when the users want to see an overview of the data, they zoom out from Main View.

- Regarding using the zoom to see the pattern versus the overview, this may have been cause by poor positioning of the overview button. The current position is not in the centre of the users’s field of vision, so they must shift their eyes to the periphery.

5.1.6 Row/Column Mode

5.1.6.1 Row/Column Mode Comments

This section contains users’ opinions on having two modes for the system.
CHAPTER 5. RESULTS AND DISCUSSION

Row/Column Mode General Comments

- Although the user liked the idea, he wanted to be able to go across rows and columns.
- One user suggested that instead of a tabbed pane (or buttons) for selecting Row and Column mode, he suggested a checkbox that would allow him to select both at the same time.

5.1.6.2 RowColumnMode Discussion

- The check box idea seems to be a good way of giving the user options. If they checked both, hovering the mouse would highlight both row and column labels. However, the user could create a new group only by selecting rows or columns, then they could load the group and try the other mode. This way they could go across both rows and columns. To enable the user to employ both the Row and Column feature, the other mode’s groups should be still accessible even if the user changes modes. This way the user could create a group that consisted of their choice of both rows and columns. (they would not be limited to one at a time).

5.1.7 General Comments on FilooT Design

This section contains the rest of the comments that were not categorized.

General Comments on the Tool

- A user suggested we build in a help system, especially for P-value and Graph View, to explain the graph relations, weights and bar meanings.
- One user suggested the system allow for scripting languages so that they have access to the data directly.

5.2 Pre-Study Questionnaire Result and Discussion

Table 5.2 shows the pre-study questionnaire (Appendix C) results that indicated the knowledge and familiarity of the users to the study domain.

As the results of table 5.2 indicate, in average, the subjects are PhD students in Bioinformatics or related fields. None of them were familiar with the VAST Challenge before. All of them had worked with at least one VA tool.
Table 5.2: pre-questionnaire study results

<table>
<thead>
<tr>
<th>Question</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current Degree Program</td>
<td>• Postdoc (has not started yet)</td>
</tr>
<tr>
<td></td>
<td>• PhD</td>
</tr>
<tr>
<td></td>
<td>• PhD</td>
</tr>
<tr>
<td></td>
<td>• M.Sc.</td>
</tr>
<tr>
<td>Area of Study</td>
<td>• Bioinformatics</td>
</tr>
<tr>
<td></td>
<td>• Computational Biology</td>
</tr>
<tr>
<td></td>
<td>• Biological Physics</td>
</tr>
<tr>
<td></td>
<td>• Bioinformatics</td>
</tr>
<tr>
<td>Previous familiarity with VAST Challenge 2010</td>
<td>• No</td>
</tr>
<tr>
<td></td>
<td>• No</td>
</tr>
<tr>
<td></td>
<td>• No</td>
</tr>
<tr>
<td></td>
<td>• No</td>
</tr>
<tr>
<td>Use of VA tools</td>
<td>• ipython (500 hours), Gnu Plot (100 hours), MATLAB (500 hours)</td>
</tr>
<tr>
<td></td>
<td>• Graphviz, many times</td>
</tr>
<tr>
<td></td>
<td>• UCSC Genome Browser, 20 hours</td>
</tr>
<tr>
<td></td>
<td>• R HeatMap, 4 hours/day, Scatter Plots</td>
</tr>
</tbody>
</table>
Chapter 6

Conclusion

6.1 Summary

Throughout this thesis, I demonstrated the process that I used to build and evaluate a tool called FilooT. A major goal for evaluating the tool was to understand the design’s capabilities for Bioinformatics expert users. Analysis of the results of the user study illustrated the design's potential for being implemented in a real Biological system. The suggested changes in design that were discussed during the previous chapter (see Chapter 5), are future guidelines for improving the design.

6.2 Limitation and Scope

The general limitations of this study as well as the scope of the thesis is discussed in this section.

- The number of sequences in the study data-set is in order of 100, and their length is in the order of 1000 nucleotide. Although I considered a number of efficient data structures and algorithms for designing FilooT, the scale of the data and the tool were not large enough generalizes to human genome data-sets which are in order of millions of nucleotides.

- The data-set is a synthesized benchmark and the level of complexity of the data is relatively low. More specifically, there is only one kind of substitution in a column.
Likewise, there is no information about the relationship between neighbouring columns (see Section 5.1.1.3).

- The final tool is the result of an iterative work. The user study provided suggestions for future directions and refinement. It did not result in another iteration because the time limitation.

- The research methodology is focused on the analysis and design justification of different aspects of FilooT. At this stage, the current research prototype still needs to improve in many aspects, and it is not comparable to established domain software tools. For this reason, I did not run any usability test on common HCI low-level tasks, nor did I relate the tool to other existing tools. The evaluation part of the methodology focuses on higher level tasks including understanding of data trends.

- The data-set/task-set are benchmarked, public and acknowledged in the literature, and some of the study participants are domain experts. Although did not occur in this study, one potential limitation was the familiarity of the participants with the data might have impacted their feedbacks. The researcher documented the answers of how familiar they were with the data-set/task-set or similar softwares from their answers of the pre-test questionnaire.

- Scope: The study results evinced that the study tasks are similar to Bioinformatics domain-specific tasks.

### 6.3 Discussion and Future Work

The main study consists of four case studies with domain experts. Running more evaluation with more domain experts as participants will help develop more aspects of the tool so it can be used in realistic situations.

The following text is a compilation of the discussion in Chapter 5 that could be used by subsequent researchers conducting a future study in this tool or design a similar solution to the same problem.

- We had two metrics for the tool: One for making P-valueView bars and one for making Graph View edge weight. The first one seems acceptable to the participants, but the second, still needs to be checked by domain experts. None of the participants were able
to help me on this because was they were not specialized in that particular domain problem. That might need collaboration with relevant domain experts.

- Although the colours of Main View are checked with VisCheck\textsuperscript{1}, it seems that the system still needs colour checking for highlighting caused by mouseovers.

- Graph View for Column mode was unusable for interacting with all the columns (in the order of 1400 columns). It is only used when the user filters most of the columns. For representing the relationship between the columns, a new visualization technique must be adopted. Graph View for Row mode (tree representation) can be kept with changing the tree layout. The tree representation of row relationships in Graph View needs a different layout so that the each node correlates to its matching row in Main View.

- Some parts that have not been implemented should be implemented and tested: adding new column in Matrix View based on existing one, hiding/unhiding rows and columns, dragging and dropping row and columns, and animated transition for the Main View shifts.

- Overviews need to show all the data, not just a larger window. To guarantee the visibility of all rows without overlaps, there should be at least one vertical overview pixel per row. If there are more rows than available pixels, some aggregation is needed. To present more than one distribution in one vertical pixel, we can calculate the minimum, average or maximum of the intensity values for that pixel \cite{43}.

- If the design ideas were finalized, they can implement them in IMAS or similar frameworks, and compared to similar softwares. The report could be a quantitative user study on task time and error rate. Usability testing could be done on low-level tasks in IMAS. After that, a quantitative study on IMAS could compare task completion time and the accuracy of the results to another similar tool. This would show the design potential of FilooT for Bioinformatic users.

- Different design alternatives that seem to work could be provided as options for the users to select between. Alternatively, the various designs could be tested in separate studies and the results applied to the related feature in the tool.

\textsuperscript{1}www.vischeck.com
Appendix A

Original Dataset and Tasks

The data consists of 56 strains of a particular original virus, which are the result of spearing of a disease over time to different infected people. Each of these strains has a gene sequence of about 1400 nucleotides with one or more nucleotide changes from the original virus sequence. The data consists of three datasets (each in separate file)

A.1 Files

In this project we use three data files used and modified from the vast mini-challenge 3 dataset: 1.CurrentOutbreakSequences.txt 2.DiseaseCharacteristics.txt 3.NativeSequences.txt

A.1.1 file1

This file contains the different viral strains genetic sequence with an identification label that collected from Drafa viral mutants during the current outbreak. Example of one viral strain sequence: 118:
ATGTCACCAGCCCTGCGCAGTTCATAGGCGCTCTCTTTCCGCAGACGGAACACGGGTCTCTTC
TGGATGGTGGGTTTGTGGGAAAGACTTGAGGCTTAACGCATAC
(size of 1400 nucleotides)

A.1.2 file2

There are some Characteristics for each of the evolved viral strains: And an explanation about the table. This table consists 56 strains information.
APPENDIX A. ORIGINAL DATASET AND TASKS

Example:

Table A.1: Sequence Characteristics Table

<table>
<thead>
<tr>
<th>ID</th>
<th>Symptoms</th>
<th>Mortality</th>
<th>Complications</th>
<th>Drug Resistance</th>
<th>At-Risk Vulnerability</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>Mild</td>
<td>High</td>
<td>Minor</td>
<td>Resistant</td>
<td>High</td>
</tr>
<tr>
<td>256</td>
<td>Severe</td>
<td>Medium</td>
<td>Major</td>
<td>Susceptible</td>
<td>Medium</td>
</tr>
<tr>
<td>19</td>
<td>Severe</td>
<td>Low</td>
<td>Major</td>
<td>Intermediate</td>
<td>Low</td>
</tr>
<tr>
<td>4</td>
<td>Moderate</td>
<td>High</td>
<td>Minor</td>
<td>Resistant</td>
<td>High</td>
</tr>
<tr>
<td>200</td>
<td>Mild</td>
<td>High</td>
<td>Major</td>
<td>Resistant</td>
<td>Low</td>
</tr>
</tbody>
</table>

Definitions:
Symptoms what a patient experiences (e.g., pain, sore throat, vomiting, swelling, tremors)
Mortality number of deaths as a result of disease
Complications unfavourable evolution of illness (e.g. deafness, spontaneous abortion)
Drug Resistance mutant vulnerability to anti viral drugs
At Risk Vulnerability disproportional effect on certain risk groups (e.g. children, elderly)

A.1.3 file3

This dataset contains genetic sequences for Drafa viral strains collected from research labs and hospitals in Africa prior to the outbreak.

A.2 Tasks

A.2.1 Task 1

What is the region or country of origin for the current outbreak? Please provide your answer as the name of the native viral strain along with a brief explanation.

A.2.2 Task 2

Over time, the virus spreads and the diversity of the virus increases as it mutates. Two patients infected with the Drafa virus are in the same hospital as Nicolai. Nicolai has a strain identified by sequence 583. One patient has a strain identified by sequence 123 and the other has a strain identified by sequence 51. Assume only a single viral strain is in each
patient. Which patient likely contracted the illness from Nicolai and why? Please provide your answer as the sequence number along with a brief explanation.

A.2.3 Task 3

Signs and symptoms of the Drafa virus are varied and humans react differently to infection. Some mutant strains from the current outbreak have been reported as being worse than others for the patients that come in contact with them.

Identify the top 3 mutations that lead to an increase in symptom severity (a disease characteristic). The mutations involve one or more base substitutions. For this question, the biological properties of the underlying amino acid sequence patterns are not significant in determining disease characteristics.

For each mutation provide the base substitutions and their position in the sequence (left to right) where the base substitutions occurred. For example,

$C \rightarrow G$, 456 (C changed to G at position 456)

$G \rightarrow A$, 513 and $T \rightarrow A$, 907 (G changed to A at position 513 and T changed to A at position 907)

$A \rightarrow G$, 39 (A changed to G at position 39)

A.2.4 Task 4

Due to the rapid spread of the virus and limited resources, medical personnel would like to focus on treatments and quarantine procedures for the worst of the mutant strains from the current outbreak, not just symptoms as in the previous question. To find the most dangerous viral mutants, experts are monitoring multiple disease characteristics.

Consider each virulence and drug resistance characteristic as equally important. Identify the top 3 mutations that lead to the most dangerous viral strains. The mutations involve one or more base substitutions. In a worst case scenario, a very dangerous strain could cause severe symptoms, have high mortality, cause major complications, exhibit resistance to anti viral drugs, and target high risk groups. For this question, the biological properties of the underlying amino acid sequence patterns are not significant in determining disease characteristics.
For each mutation provide the base substitutions and their position in the sequence (left to right) where the base substitutions occurred. For example,

\[ C \rightarrow G, \ 456 \text{ (C changed to G at position 456)} \]
\[ G \rightarrow A, \ 513 \text{ and } T \rightarrow A, \ 907 \text{ (G changed to A at position 513 and T changed to A at position 907)} \]
\[ A \rightarrow G, \ 39 \text{ (A changed to G at position 39)} \]
Appendix B

VAST Challenge 2010 Solutions

B.1 VAST Challenge 2010 Mini Challenge 3, Task 3 Solution

VAST Challenge 2010 Official Solution

Table B.1: Top mutations based on correlation to symptom severity

<table>
<thead>
<tr>
<th>Position</th>
<th>Base Substitution</th>
</tr>
</thead>
<tbody>
<tr>
<td>946</td>
<td>A → T</td>
</tr>
<tr>
<td>842</td>
<td>T → C</td>
</tr>
<tr>
<td>269</td>
<td>A → C</td>
</tr>
<tr>
<td>223</td>
<td>A → G</td>
</tr>
<tr>
<td>212</td>
<td>G → C</td>
</tr>
</tbody>
</table>

ManyNets Solution

T → C, 842 and A → T, 955 (path from 531 to 583)
A → C, 161 and A → G, 223 and T → C, 790 (path from 531 to 952)
A → C, 269 and A → T, 843 (path from 531 to 99)

Cognizant BFS Innovations prototype Solution

A → C, 269
G → C, 212
A → T 946
GeneTracer Solution

A→C, 269
A→T, 946 and T→C, 842
A→G, 223

SequenceView Solution

A→C, 269 - severe x4
A→T, 946 - severe x9; moderate x5 (842 is very correlated)
G→C, 212 - severe x3; moderate x1.

B.2 VAST Challenge 2010 Mini Challenge 3, Task 4 solution

VAST Challenge 2010 Official Solution

Table B.2: Task 4 Official solution

<table>
<thead>
<tr>
<th>Position</th>
<th>Base Substitution</th>
</tr>
</thead>
<tbody>
<tr>
<td>22</td>
<td>G → C</td>
</tr>
<tr>
<td>79</td>
<td>C→ A</td>
</tr>
<tr>
<td>946</td>
<td>A → T</td>
</tr>
<tr>
<td>842</td>
<td>T → C</td>
</tr>
<tr>
<td>269</td>
<td>A → C</td>
</tr>
<tr>
<td>223</td>
<td>A → G</td>
</tr>
<tr>
<td>790</td>
<td>T → C</td>
</tr>
</tbody>
</table>

ManyNets Solution

A → C, 269 and T → C, 842 and A → T, 946 (strain 123)
A → C, 269 and T → C, 527 and T → C, 842 and A → T, 946 (strain 118)
A → C, 170 and T → C, 842 and G → C, 848 and A → T, 946 (strain 501)

Cognizant BFS Innovations prototype Solution

A→C, 269
G→C, 223
A→T, 946
GeneTracer Solution

A → T, 946 and T → C, 842
T → C, 790
A → G, 223

SequenceView Solution

T → C, 842 - well correlated with 946. Strongly affects vulnerability
T → C, 790 - not as severe as 842, but affects different DNA sequences. Complimentary to the mutation above
A → T, 955 - only 4 base mutations and not particularly severe, but complementary to both mutations above (affect different strains)
Appendix C

Study Documents

- Recruitment Advertisement
- Informed Consent
- Pre-Study Questionnaire
- Task Description
- Compensation form
Recruitment Ad.

Data Viewing Study
Call for Participation

My name is Mahshid Zeinaly. I am a master’s student at School of Interactive Arts and Technology, Simon Fraser University working on an information visualization tool. My thesis research is focused on designing a new tool for viewing a public synthesized health data as well as understanding the design choices’ strengths and pitfalls in solving some pre-defined tasks. We are looking for participants to help by performing the study’s tasks on the tool in a lab setting.

You will be asked to use the system to solve the task problems. You will be encouraged to think aloud, express your thoughts and ask any questions during the study. At the end, we will have an interview about your experience with the tool. This study will take less than 1.5 hours.

We are looking for undergraduate/graduate/post-doctorate students over the age of 19 who are familiar with DNA multi-alignment concept in bioinformatics. If you are interested in participating, please contact me at [redacted]

Beyond the basic knowledge of bioinformatics DNA multiple sequence alignment, and age, participants of all genders, ethnicities, abilities, and familiarity levels with biological concepts/information visualization tools are encouraged to participate!

Although the bioinformatics knowledge is mandatory for participating, this study is not to test knowledge of the subject, but it is to evaluate the tool we have designed.

Typical compensation for participation is $10 (for one-hour study). If it goes over that time it is $15 (this study will take less than 1.5 hours.)

Sincerely,

Mahshid Zeinaly, [redacted]
School of Interactive Arts and Technology, Simon Fraser University
INFORMED CONSENT

This research is being conducted under the permission of the Simon Fraser Research Ethics Board. Simon Fraser University and those conducting this study subscribe to the ethical conduct of research and to the protection at all times of the interests, comfort, and safety of participants. This form and the information it contains are given to you for your own protection and to ensure your full understanding of the procedures, risks, and benefits described below.

Should you wish to obtain information about your rights as a participant in research, or about the responsibilities of researchers, or if you have any questions, concerns or complaints about the manner in which you were treated in this study, please contact the supervisor of this study, Dr. Chris D. Shaw by email at [email protected], or secondary to Dr. Hal Weinberg, the Director, Office of Research Ethics by email at [email protected], or phone at 778-782-6593.

TITLE: Data Viewing Study

PRINCIPAL INVESTIGATOR:

Mahshid Zeinaly, School of Interactive Arts and Technology, Simon Fraser University, Email: [email protected],

SUPERVISOR:

Chris D. Shaw, School of Interactive Arts and Technology, Simon Fraser University [email protected],

PURPOSE AND BENEFITS

We have designed an information visualization tool for analyzing data by providing different views and visualizations of the whole data set. The research is designed for the primary investigator’s masters’ thesis. We are interested to learn your views on the design choices for creating this tool. The purpose of this study is to understand the strengths and weaknesses of the design choices for the different aspects of the data. The result of the study would be used to critique and improve the design or suggest improvements as a future work direction. The benefits of study are to the development of new knowledge about the designed tool.

PROCEDURE

The purpose of this experiment is to gather information on how participant can use our new tool to answer a series of questions about a dataset. The dataset consists of a public synthetic data of the mutations of a disease that is modified for the purpose of this study. The original dataset and questions come from IEEE VAST 2010 Challenge, miniChallenge3. The tool is a visualization software that visualizes the different aspects of the dataset in different views. First you will fill out a questionnaire with a few questions about your familiarity of the bioinformatics domain and your experience with similar visualization tools. You will then be trained for 10 minutes to learn
to use the basic features of the software. Next you will be given 2 questions to answer using our software. All answers to the questions can be found within the dataset. You will have 30 minutes to use the tool and find the answers. You are encouraged to write down the answers or anything that you found in the data. It would be also great if you express your thoughts and feelings during the study time. I will be using pen and paper to take notes of your actions, and your shared thoughts during the study. You could ask any questions or difficulties that you may encounter during study. When you have completed 30 minute question-answering task, you will participate in an interview in which I will ask open-ended questions about your experience with the software.

---

**RISKS**

There are no foreseeable risks to participating.

---

**CONFIDENTIALITY**

The identities of all people who participate will remain anonymous and will be kept confidential. After your participation, all electronically recorded data will be immediately stored on a secure server located in the secure server room at SFU Surrey and the papers will be placed in a locked filing cabinet. After the study, identifiable data will be stored securely on a CD-ROM in a locked cabinet. All data from individual participants will be coded so that their anonymity will be protected in any project reports and presentations that result from this work. All of the collected data will be destroyed after 6 years.

---

**CONSENT**

Your signature on this form will signify that you have received the aforementioned information in this document which describes the procedures, possible risks, and benefits of this research study, that you have received an adequate opportunity to consider the information in the document, and that you voluntarily agree to participate in the study. We are very grateful for your participation.

My participation is entirely voluntary and I may refuse to participate or withdraw from the study at any time and it will have no adverse effects on my grades or evaluation in my courses. I may obtain copies of the results of this study, upon its completion by contacting Dr. Chris Shaw by email at (…@…).

I certify that I understand the procedures to be used and that I understand this document, and that I have been able to receive clarification of any aspects of this study about which I have had questions.

Print Name: ______________________________________

Signature: __________________________________________________________

Date (use format Month / Day / Year): ____________________________

Page 2 of 2
PRE-STUDY QUESTIONNAIRE

SFU Ethics Approval Application [application #] Data Viewing Study
Investigator Name: Mahshid Zeinaly Baraghoush
Supervisor: Chris D. Shaw
Investigator Department: School of Interactive Arts and Technology

Highest Degree you have obtained: ________________
Your current degree program, if any: ________________
Your Area of Study: ______________________________

Are you familiar with the VAST challenge 2010 mini-challenge 3 dataset: If so, please explain how much you worked with that data:
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________

Do you use any Visual Analytics or Information Visualization tools for your work/research? If so, please name them and estimate the number of hours you have used each:
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
**STUDY TASK/DATA DESCRIPTION**

SFU Ethics Approval Application [ORE 2012s0710] Data Viewing Study  
Investigator Name: Mahshid Zeinaly Baraghoush, [REDACTED]  
Supervisor: Chris D. Shaw, [REDACTED]  
Investigator Department: School of Interactive Arts and Technology

The dataset consists of 56 strains of a particular original virus, which are the result of spearing of a disease over time to different infected people. Each of these strains has a gene sequence of 1400 nucleotides with one or more nucleotide changes from the original virus sequence. There are some Characteristics for each of the evolved viral strains: And an explanation about the table. This table consists 56 strains information.

Below is a sample data.

<table>
<thead>
<tr>
<th>Sequence ID</th>
<th>Symptoms</th>
<th>Mortality</th>
<th>Complications</th>
<th>Drug Resistance</th>
<th>At-Risk Vulnerability</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>Mild</td>
<td>High</td>
<td>Minor</td>
<td>Resistant</td>
<td>High</td>
</tr>
<tr>
<td>256</td>
<td>Severe</td>
<td>Medium</td>
<td>Major</td>
<td>Susceptible</td>
<td>Medium</td>
</tr>
<tr>
<td>19</td>
<td>Severe</td>
<td>Low</td>
<td>Major</td>
<td>Intermediate</td>
<td>Low</td>
</tr>
<tr>
<td>4</td>
<td>Moderate</td>
<td>High</td>
<td>Minor</td>
<td>Resistant</td>
<td>High</td>
</tr>
<tr>
<td>200</td>
<td>Mild</td>
<td>High</td>
<td>Major</td>
<td>Resistant</td>
<td>Low</td>
</tr>
</tbody>
</table>

**DEFINITIONS**

Symptoms: what a patient experiences (e.g., pain, sore throat, vomiting, swelling, tremors)  
Mortality: number of deaths as a result of disease  
Complications: unfavorable evolution of illness (e.g. deafness, spontaneous abortion)  
Drug Resistance: mutant vulnerability to anti viral drugs  
At Risk Vulnerability: disproportional effect on certain risk groups (e.g. children, elderly)

**TASK 1:**

Identify mutations that lead to an increase in symptom severity (a disease characteristic)  
Example of the output that is the Severity-driven substitutions in the order of importance:  
2. C->G, 93

**TASK 2:**

Identify mutations that lead to the most dangerous viral strains.

**TASK 3:**

Nicolai has a strain identified by sequence 583. One patient has a strain identified by sequence 123 and the other has a strain identified by sequence 51. Which patient contracted the illness from Nicolai and why?

**NOTE:**

Using the tool you are able to see different aspects of the dataset. This study is designed to test how much this tool is helpful for finding the answers and it is not your knowledge that is being tested. The process of finding the answers is more important than the accuracy of the result. The researched is taking notes while you are using the system. She is available to answer your questions at any time. Feel free to speak loud or express any thoughts/suggestions or confusions during the study.
## Compensation Record

<table>
<thead>
<tr>
<th>Date</th>
<th>Participant Name</th>
<th>ID</th>
<th>Paid</th>
<th>Participant Signature</th>
<th>Participant E-mail</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Appendix D

IMAS Documentation

D.1 System Overview and Tools

In order to document IMAS system, I used two tools: Dyoxygen and Visual Paradigm.

D.2 Feature-related Documentation

D.2.1 Create Feature
D.2.2 Hide Feature
D.2.3 Delete Feature

D.2.3.1 UseCases
D.2.3.2 Sequence Diagrams

D.3 Pairwise-alignment Documentation

D.3.1 Sequence Diagrams
D.3.2 Compuse Structure Diagrams
D.3.3 Object Diagrams
D.3.4 Activity Diagrams
### Table D.1: View Selected Subseq. Menu Options: UseCase Description

<table>
<thead>
<tr>
<th>Primary Actor</th>
<th>Analyzer</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description</strong></td>
<td>The analyzer wants to view what are her options that she could do on a selected subsequence</td>
</tr>
<tr>
<td><strong>PreCondition</strong></td>
<td>A subsequence has already been selected. Trigger: Right Click on a subsequence of a primary Sequence.</td>
</tr>
</tbody>
</table>
| **Basic Flow** | 1. The user right clicks on a primary sequence (NT or AA area)  
2. The system checks if a valid subsequence has been selected  
   If a valid subsequence is being selected  
      A menu pops up  
   else  
      Nothing happens |
| **Alternate Flows** | Nothing happens on the right click! |
| **EndCondition** | The menu pops up |

### Table D.2: Create Feature: UseCase Description

<table>
<thead>
<tr>
<th>Primary Actor</th>
<th>Analyzer</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description</strong></td>
<td>The analyzer wants to create a new feature out of her selected subsequence</td>
</tr>
<tr>
<td><strong>PreCondition</strong></td>
<td>A menu for a selected subsequence has been popped up Trigger: Clicking on the &quot;Create Feature&quot; from the menu</td>
</tr>
</tbody>
</table>
| **Basic Flow** | 1. The use clicks on a the "create feature" from the menu  
2. The system adds a new feature to the existing features  
3. The new feature is shown in the ORF followed this strategy:  
   If there is a spot available in between of the current features  
      Place the new feature into that spot  
   else  
      Increase the size of ORF by add a new row at the buttom and place the feature into that new area |
| **Alternate Flows** | Nothing happens on the right click! |
| **EndCondition** | The feature created, added to feature lists and is shown on the ORF |
This is when the user clicked on a primary sequence and has the view of NT and AA sequence. The user then can select a subsequence and then select to view what are her options for further actions. She also can create a feature out of the selected subsequence.

Figure D.1: Menu Options usecase.
Figure D.2: Sequence Diagram for ViewSubseqMen.
Figure D.3: Sequence Diagram for "Create Feature" UseCae
Figure D.4: Sequence Diagram for OrfContainer.AddFeatureBoxes() : a Ref in Create Feature SeqDiagram
Figure D.5: Sequence Diagram for FeatureBox.Draw() : a Ref in Create Feature SeqDiagram
Figure D.6: Sequence Diagram for Hide Feature.
Figure D.7: UseCase for Delete Featuree
Figure D.8: Sequence Diagram for ViewFeatureMenu
Figure D.9: Sequence Diagram for DeleteFeature
Figure D.10: Sequence diagram for DeleteFeature Ref OrfContainer DeleteFeatureBox
Figure D.11: Sequence Diagram for Blast Against Nucleotides.
Figure D.12: Sequence Diagram for PairwiseAlignments.
Figure D.13: Sequence Diagram for Blast Output
Figure D.14: Compose Structure Diagram for Blast HSP Rect.
Figure D.15: Object Diagram for showing an example of Result Container.
Figure D.16: Activity Diagram for Result Container “DoLayout” algorithm).
D.4  Multi-alignment Documentation

D.4.1  UseCases

D.4.2  Sequence Diagrams

D.4.3  Compose Structure Diagrams
Figure D.17: Activity Diagram for Sequence Box “Draw” algorithm.
Figure D.18: Activity Diagram for ManagedRect “OverlapXanotherManagedRect” algorithm.
Figure D.19: Activity Diagram for ManagedRectLayout “ManagedRect aNewChild” algorithm.

APPENDIX D. IMAS DOCUMENTATION

133
Figure D.20: ActivityDiagram for ManagedRect “Layout”.

- Lay out all children within this (parent) box.
- Some rect is calling the "Layout" method (messy rect).

1. Retrieve all the children. 
   `ManagedRect::GetChildren(); GetChildren(); children`

2. For each child:
   - `kids.push_back(child)`

3. `kids` contains all the children.

4. Delete children
   `ManagedRect::children.clear()`

5. For each kid from `kids`:
   - `kid` (messy within its parent)
   - Layout(kid) ref to Activity Diagram for ManagedRect.Layout(ManagedRect *child)

6. `kid` (ordered within its parent)

7. Now `ManagedRect` could accept this `kid` as its children!
   - `ManagedRect::skeleton->AddChildRect(*kid)`

8. Rect [All the children is in their own good places]

- Lay out 1 child within this (parent) box.
Figure D.21: UseCase for Multiple Sequence Alignment in IMAS.
Figure D.22: Sequence Diagram for NTPane Update All Views.
Figure D.23: Sequence Diagram for Multi Alignment.
Figure D.24: Sequence Diagram for Multi Alignment MenuPop.
Figure D.25: Sequence Diagram for MultiAlignBlastHits.
Figure D.26: Compose Structure Diagram for MultiAlignOutputRect.
Bibliography


Index

Information Visualization, 33, 76, 77
Interaction, 2, 3, 26, 36–38, 42, 43, 65, 77, 79, 80, 83–85
Interaction Design, 2, 42, 76
Interaction Techniques, 36

Task Translation, 41

User Experience, 78, 79

VAST Challenge, 61, 85, 91, 93
Visual Analytics, 1, 3, 85
Visual Encodings, 42