2 Requirement Specification

2.1 Introduction

Particle Filter Simulator is a lightweight application which is mainly used by college students and teachers. Because Particle Filter Simulator is aimed for teaching of image and data processing, thus, in order to meet the main needs of users and provide positive user experience, there are a large amount of significant requirements should be carefully considered and implemented. In this part of report, it will illustrate the process of gathering and implementing requirement and provide a list of different types of requirements. For each specific requirement, it will be analyzed and explained in more details.

2.2 User case Diagram

User case diagram (Figure 8) could describe how users interact with system and it could be the first step from requirement specification to software implementation. It could help developers implement the main function of Particle Filter Simulator and provide a clear diagram for users to understand each part of the software.

Figure 8: The actors, use cases and system boundary required in Particle Filter Simulator system.

2.3 Prototype

Software prototype provides a brief user interface and it has divided the whole software into several parts. Developer could better implement division and cooperation by focusing on each different component in the prototype. It also provides a feasible option to validate if the software design achieves expected user experience.

1. Start interface (Figure 9): After the software is opened by user, user should first select different algorithms. It could be the interface that users firstly enter after the software is running and help users to get start easily.

Figure 9: Start interface

1. Navigating bar: Navigating bar is placed on top of the window and it contains three main options:

1. File option (Figure 10) include the function of operating input and output file.

2. View option (Figure 11) include the function of setting window on the top or changing the software appearance.

3. Help option (Figure 12) include the help document and software update.

Figure 10: Navigating bar-File

Figure 11: Navigating bar-View

Figure 12: Navigating bar-Help

1. Main interface (Figure 13, 14 & 15): Main interface of generating data and image, which includes three components: function button, outcome image and parameter controller. Function button includes several functional buttons to deal with data and image. Outcome image includes an area to generate a plot image with coordinate axis and two scroll bars. Parameter controller contains a list containing parameter names, several sliders used to adjust parameter value and a start and refresh button.

Figure 13: Main interface-preparing

Figure 14: Main interface-executing

Figure 15: Main interface-complete

2.4 Requirements list

During the procedure of software design and implementation, developers found that some requirements are difficult to implement and there are also some new requirements need to be considered. In this section, it mainly focusing on the requirements that has been appeared in the software design process and no matter whether they have been implemented or not.

2.4.1 Functional requirements

Particle Filter Simulator is using the algorithm selected by the user to generate the outcome image. These are the main functional requirements appearing during the software design process:

1. The software should illustrate the correctly usage method in help document.

2. The software should provide the function of selecting different algorithms

3. The software should display the brief explanation of algorithms when user is selecting an algorithm.

4. The software should provide some existing sample data to users without other available data.

5. The software needs check the format of the file type and give user hints when users import specific sample data.

6. The software should use the algorithm which is selected by user to deal with the import sample data and generate the outcome image.

7. The software should provide modifiable parameter list and brief explanations of each parameter.

8. The software should have the function of modifying parameters which are initially set as default value.

9. The software should implement the function of exporting images, data, or both of them according to the users’ decision.

10. The software should permit users to combine two images together to compare the differences between them.

11. The software should provide the function of clearing the current data and results if users want to import another sample data or select another algorithm.

12. The software should provide a function of putting the software window on the top.

13. A restart button should be provided to handle the situation of software crashing.

14. The software need to check version update automatically and submit bugs which are collected by users to software maintainer.

15. When the software is using algorithm to generate image, a progress bar or a waiting hint message should be provided which is used to display the running state of software.

16. The range of parameters should be limited to avoid the situation that increasing running time of algorithm and software crashing.

2.4.2 Non-functional requirements

Performance, Security & Environment: The non-functional requirements are given as follow which include the environment requirements, performance requirements and data requirements: 1. The window size of the software should be suitable for demonstration.

2. The user interface should be convenient for user activity.

3. The running time of using the algorithm to deal with the sample data should be as quick as possible, for instance, it should be less than 5 seconds with the consideration of the computability of Javascript.

4. The system resources occupied by software should be limited.

5. The software needs to run on major platforms and operating systems.

6. The software should be convenient for update and extension.

7. The outcome image should have expected accuracy compare with Matlab when the range of several parameters is limited.

8. The software requires appropriate software capacity to be convenient for users to download and use in a short period of time.

9. It should be convenient to passing parameter data between each component in different parts of the software.

10. The software is free, it is better that the development and maintenance costs should be properly reduced.