Reflection and Traceability Report on Image Feature Correspondences for Camera Calibration

Kiran Singh

[Reflection is an important component of getting the full benefits from a learning experience. Besides the intrinsic benefits of reflection, this document will be used to help the TAs grade how well your team responded to feedback. Therefore, traceability between Revision 0 and Revision 1 is and important part of the reflection exercise. In addition, several CEAB (Canadian Engineering Accreditation Board) Learning Outcomes (LOs) will be assessed based on your reflections. —TPLT]

1 Changes in Response to Feedback

[Summarize the changes made over the course of the project in response to feedback from TAs, the instructor, teammates, other teams, the project supervisor (if present), and from user testers. —TPLT]

[For those teams with an external supervisor, please highlight how the feedback from the supervisor shaped your project. In particular, you should highlight the supervisor's response to your Rev 0 demonstration to them. —TPLT]

[Version control can make the summary relatively easy, if you used issues and meaningful commits. If you feedback is in an issue, and you responded in the issue tracker, you can point to the issue as part of explaining your changes. If addressing the issue required changes to code or documentation, you can point to the specific commit that made the changes. Although the links are helpful for the details, you should include a label for each item of feedback so that the reader has an idea of what each item is about without the need to click on everything to find out. —TPLT]

[If you were not organized with your commits, traceability between feedback and commits will not be feasible to capture after the fact. You will instead need to spend time writing down a summary of the changes made in response to each item of feedback. —TPLT]

[You should address EVERY item of feedback. A table or itemized list is recommended. You should record every item of feedback, along with the source of that feedback and the change you made in response to that feedback. The response can be a change to your documentation, code, or development process. The response can also be the reason why no changes were made in response to the feedback. To make this information manageable, you will record the

feedback and response separately for each deliverable in the sections that follow. —TPLT]

[If the feedback is general or incomplete, the TA (or instructor) will not be able to grade your response to feedback. In that case your grade on this document, and likely the Revision 1 versions of the other documents will be low. —TPLT]

1.1 SRS and Hazard Analysis

Hazard Analysis was determined to exceed the scope of work for the development of the Image Feature Correspondences for Camera Calibration software. This software will not be used in real-time or safety critical applications.

Version 1.0 of the **SRS** was published under commit **8776a2b**. Version 2.0 of the **SRS** was published under commit **d139755**.

Revision Log of Document Updates

- **Source of Feedback:** GitHub Issue #2
  Issue Identified: The *Table of Units* was not applicable to the software system.
  Action Taken: The table was removed from the document.
  Commit: c7d6830

- **Source of Feedback:** GitHub Issue #2
  Issue Identified: Inconsistent symbol definitions across theoretical models, instance models, and data definitions.
  Action Taken: The *Table of Symbols* was updated for consistency across all sections.
  Commit: 6c49512

- **Source of Feedback:** GitHub Issue #4
  Issue Identified: Expected user inputs and responsibilities were unclear in the System Context Diagram.
  Action Taken: Diagram and descriptive text were revised to clarify user input responsibilities.
  Commit: 86e483c

- **Source of Feedback:** Instructor (verbal) during VnV Plan discussion
  Issue Identified: Lack of documentation for OpenCV-based constraints.
  Action Taken: A new system constraint based on OpenCV limitations was added.
  Commit: c7d6830

- **Source of Feedback:** GitHub Issue #2
  Issue Identified: Definition of extrinsic parameters was vague.
  Action Taken: Definition of extrinsic parameters was clarified.
  Commit: e0236d3

- **Source of Feedback:** GitHub Issue #2
  Issue Identified: Requirement GS5 was outdated and unclear.
  Action Taken: Replaced GS5 with new requirement R15 for feature correspondence reporting.
  Commit: 6c9a58a

- **Source of Feedback:** GitHub Issue #2
  Issue Identified: Requirements GS1 and GS2 were no longer needed.
  Action Taken: GS1 and GS2 were removed.
  Commit: e0236d3

- **Source of Feedback:** GitHub Issue #2
  Issue Identified: Theoretical and instance models used inconsistent or unclear notation.
  Action Taken: TM01–TM04 and IM01–IM04 were revised for consistent mathematical presentation.
  Commit: 6c49512

- **Source of Feedback:** GitHub Issue #2
  Issue Identified: The definition of the XOR operation in TM4 was unclear.
  Action Taken: Improved definition and symbolic notation for XOR was added.
  Commits: c7d6830, 6c49512

- **Source of Feedback:** GitHub Issue #2
  Issue Identified: IM01 lacked detailed steps for computing the Gaussian-smoothed image.
  Action Taken: IM01 was expanded to include a full description of the required computation steps.
  Commit: 6c49512

- **Source of Feedback:** GitHub Issue #2
  Issue Identified: No non-functional requirements for software quality were provided.
  Action Taken: Added five non-functional requirements (NFR1–NFR5) covering reliability, usability, maintainability, and performance.
  Commit: 6c9a58a

- **Source of Feedback:** GitHub Issue #5
  Issue Identified: Table of Symbols was not consistent with mathematical models.
  Action Taken: Table of Symbols was updated to align with definitions and notation in theoretical and instance models.
  Commit: 6c49512

1.2 Design and Design Documentation

1.3 VnV Plan and Report

2 Challenge Level and Extras

2.1 Challenge Level

[State the challenge level (advanced, general, basic) for your project. Your challenge level should exactly match what is included in your problem statement. This should be the challenge level agreed on between you and the course instructor. —TPLT]

2.2 Extras

[Summarize the extras (if any) that were tackled by this project. Extras can include usability testing, code walkthroughs, user documentation, formal proof, GenderMag personas, Design Thinking, etc. Extras should have already been approved by the course instructor as included in your problem statement. —TPLT]

3 Design Iteration (LO11 (PrototypeIterate))

[Explain how you arrived at your final design and implementation. How did the design evolve from the first version to the final version? —TPLT]

[Don't just say what you changed, say why you changed it. The needs of the client should be part of the explanation. For example, if you made changes in response to usability testing, explain what the testing found and what changes it led to. —TPLT]

4 Design Decisions (LO12)

[Reflect and justify your design decisions. How did limitations, assumptions, and constraints influence your decisions? Discuss each of these separately. —TPLT]

5 Economic Considerations (LO23)

[Is there a market for your product? What would be involved in marketing your product? What is your estimate of the cost to produce a version that you could sell? What would you charge for your product? How many units would you have to sell to make money? If your product isn't something that would be sold, like an open source project, how would you go about attracting users? How many potential users currently exist? —TPLT]

6 Reflection on Project Management (LO24)

[This question focuses on processes and tools used for project management. —TPLT]

6.1 How Does Your Project Management Compare to Your Development Plan

[Did you follow your Development plan, with respect to the team meeting plan, team communication plan, team member roles and workflow plan. Did you use the technology you planned on using? —TPLT]

6.2 What Went Well?

[What went well for your project management in terms of processes and technology? —TPLT]

6.3 What Went Wrong?

[What went wrong in terms of processes and technology? —TPLT]

6.4 What Would you Do Differently Next Time?

[What will you do differently for your next project? —TPLT]

7 Reflection on Capstone

[This question focuses on what you learned during the course of the capstone project. —TPLT]

7.1 Which Courses Were Relevant

[Which of the courses you have taken were relevant for the capstone project? —TPLT]

7.2 Knowledge/Skills Outside of Courses

[What skills/knowledge did you need to acquire for your capstone project that was outside of the courses you took? —TPLT]