Mobile Vision Based Risk Detection: Using computer vision to detect potential risks in a Parrot AR.Drone 2.0

Report Name	Methodology for the project
Author (User Id)	Joseph James (jgj2)
Supervisor (User Id)	Myra Wilson (mxw)
Module	CS39440
Degree Scheme	G400 (Computer Science)
Date	February 11, 2016
Revision	1.0
Status	Release

---

1 Introduction

For this project, a methodology will have to be followed. This is in order to keep the project on track and keeping forward progress. This document will introduce the initial methodology, explaining the thoughts behind any decisions and explaining any artifacts that will be produced.

The initial methodology will draw mainly from scrum and kanban, taking an iterative approach to development and process improvement from scrum and using a kanban workflow for use throughout the sprints.

Sprints are being used so that constant improvement can be made to the process being used, as well as allowing for artifacts like burn-down charts to be introduced. The kanban system is used during the sprint to ensure work is done effectively and fully tested. It also forces a limit and provides a motivation to finish all the tasks for the sprint, which should help improve work rate.

2 Overarching methodology

Sprints Each development iteration will take place in a sprint, with each sprint lasting 2 weeks. During this time, a particular number of stories should be completed from the sprint backlog and if they're not, they'll be added back onto the project backlog.

The 2 week sprint will allow for supervisory meetings at the end and beginning of a sprint, as well as a mid-sprint meeting. At the beginning of the sprint, the meeting can discuss what is going to be done during the week and any issues that are going to effect the upcoming sprint. This same meeting would also be used for the end of each sprint, so issues that affected the previous sprint can start to be resolved or discussed and the supervisor can be made aware of the progress that has been made.

The mid-sprint meeting would be used to discuss issues with work that is currently happening, how it's going and to also get help with anything that is particularly difficult. A mid-sprint meeting also means that issues can be solved before the next sprint, stopping slowing down of progress on the project.

Kanban work flow This will be used with the stories for the current sprint and will have the following value stream:

Sprint Backlog Contains all the stories from the current sprint that have not yet been completed. At the start of this process, 10 stories will be added to each sprint backlog. This is an arbitrary number, which will be altered depending on the amount of work that is necessary to be undertaken throughout the project.

Active Stories This will contain the stories currently being worked on, with an initial work in progress limit of 4. This number is arbitrary and will likely change during the process if either too much work is being done, or not enough.

Testing Stories that have been completed will then move into the testing stage of the value stream, which is also included in the WIP limit. For example, 2 stories can be active and 2 can be in testing but no more may be moved into the active stories until

the stories in the testing stage are done. Having this stage will ensure proper testing of all parts of the system, as if something does not pass a test it will be moved back into the active stories category.

Completed Any stories that are finished and have passed all tests will be held in the completed stage. These stories shouldn't need to be touched again once they've been moved into this stage and should only be moved back out if it's absolutely necessary.

Sprint Review As this is a single person project, it will be impossible to have a discussion with peers about how the sprint itself went. However, at the end of each sprint, a small document discussing the sprint will be produced. In this, any problems with the sprint itself will be highlighted, for example if the WIP limit was too small so it slowed down progress or too large, so the work was overwhelming.

After the discussion of the issues, a potential solution will be chosen and noted in the document. This solution will be implemented in the next sprint and this will allow for an evolving method, which will iteratively improve. This will also allow for documentation of the changes in the process, which will be useful for the final report.

Progress Review Again, with no peers working on this project, it will be impossible to have a scheduled meeting to discuss progress made. The solution to this problem remains the same however, in that a document will be produced detailing work that was done, issues that came up in the work and if there were issues, how they'll be resolved.

This document will be similar to the sprint review, however it will also allow for critical analysis of how effective the process is. The amount of items completed in the progress review will correlate with the effectiveness of the process used, and as the process improves, the number of completed items will likely fluctuate around a peak efficiency.

3 Artifacts

Project Backlog This backlog will contain all the stories for the project, as progress is made and more is made clear, stories will be added to this backlog, to be completed in future sprints.

Sprint backlog This backlog will contain all the stories to be completed during the current sprint. At the time of this document, it is limited to 10 stories per sprint, however this will change depending on the results of both the progress review and process review.

Kanban Board This is the board that all the stories will be held on. This could also be known as the information radiator. It is likely there will be both a manual version, made with string and sticky notes, and a digital version, so it is always accessible.

Progress/Process Review Documents These documents will be produced constantly throughout the project, providing a retrospective on how the process has changed and what progress has been made. These are going to be stored digitally, in the version control.