Sprint 1 - Writeup

Behnam Saeedi, Margaux Masson, Kamilla Aslami, Nghi Duong, Dhruv Jawalkar
CS561: Software Engineering Methods
Due Oct 21st 11:59pm
Fall 2018
Repository: https://github.com/BeNsAeI/PADSR

CONTENTS

1 INTRODUCTION

Purpose of this project is to produce a system that creates 3D videos using normal camera footage. Our repository is available at the following link(https://github.com/BeNsAeI/PADSR). This is achieved by using certain camera tricks and giving the illusion of having stereo camera setup by exploiting the parallax effect. In this sprint we intended to introduce some of the tools which we will be needing in order to realize this project. In this sprint we take three important steps. First step is to take in two images that are some distance apart and create a depth map of the scene. The second stem is to recognize which direction the camera is moving based on a video. The final step is to take a video and convert it to a depth map video.

2 GITHUB REPOSITORY INSTRUCTIONS

The repository is located at "https://github.com/BeNsAeI/PADSR" and contains many different folders. The files that we developed for the purpose of this project are all located in the Integration folder. Furthermore, We can find all of our documentation in the Documentation folder. We also have few other folders in the repository that are meant for testing, development and figuring out what is achievable. Likewise, in the documentation folder we have a folder for user stories. The User stories for this sprint and potential user stories for any future development are located in "Documentation/User stories/ PAD".

3 MEET THE TEAM

The following is the least of our group members and their expertise and background.

- **Behnam Saeedi (Product Owner):** Behnam Saeedi is a master student with AI and computer graphics options. His skills are mainly in embedded programming, machine learning, image processing and computer graphics.
- **Margaux Masson:** Margaux Masson is an exchange Computer Science's student at OSU from France. Her school back in France is CPE Lyon Engineering school. Her major is Computer Vision, so all the fields related to image processing and computer graphics. In OSU, she is taking the Computer Graphics, Cyber Security and Software Development Methods' courses.
- **Nghi Duong:** Nghi is a new master student concentrating computer vision and 3D modeling. His background is in mathematics and computer vision. He joined the team later than other members. (He first met the team on Oct 3rd.)
- **Kamilla Aslami (Scrum Master):** Kamilla is a first year Master's student participating in the Software Innovation Track. She is interested in Machine Learning and has used Python for 3 years in a professional setting.
- **Dhruv Jawalkar:** Dhruv is a master's student at OSU with an interest in Computer Vision, has experience in programming in Python, training neural nets to do simple vision tasks like Classification, Localization and Object Detection.

4 USER STORIES

Here is the list of User stories which we attempted to accomplish. Total allocated time for these 4 user stories were around 95 hours. We kept 10 hours for cases where one of the user stories required more attention. In our

case that ended up being the spike user story. In this section we will cover our user stories and also comment on why we decided such specifications, priorities and estimated completion times for each one. Please note, our actual user stories are concise and do not include in depth explanations on them. These user stories could be viewed on the "PAD-SR GitHub: Documentation/User stories/ PAD/" path. In this write up, texts that are written in **bold** are the texts that are in our user story 5×3 cards.

4.1 Spike (Process requirement):

Take time as a team to set everything up

This step was added to make all of our team members setup and get familiar with the process, APIs and tools we will be using for the project.

4.1.1 Priority

High: Next 3 User stories depend on this task.

4.1.2 Estimated time

30: Setting up testing and learning OpenCV is very time consuming and all members need to do this.

4.1.3 Details

OpenCV requires a setup time, furthermore, some of the members need to familiarize themselves with the environment in which we will be developing in.

4.1.4 Tasks

- Discuss test case.
- Set up OpenCV and OpenCV_Contrib (2)
- Run and study OpenCV examples (2)
- Discuss, research and study GLSL (2)

4.2 Image pair to depth map (Functional, Product, Derived requirement):

Take a pair of images and convert the to depth map images

4.2.1 Priority

High: Third User story heavily depends on this feature working properly.

4.2.2 Estimated time

20: This is a challenging task however, it is well documented.

4.2.3 Details

Given two images we need to estimate a depth image out of the two pictures

4.2.4 Tasks

- Unit test (5)
- Create depth-map (10)
- Check for potential optimization (5)

4.3 Detect motion direction in a footage (Derived requirement):

Detect if camera is moving to the left or to the right.

4.3.1 Priority

High: Third User story heavily depends on this feature working properly.

4.3.2 Estimated time

20

4.3.3 Details

We need to know given image 1 and 2, whether image 1 is on the left or right side of image 2

4.3.4 Tasks

- Unit test(4)
- Setup function and OpenCV pipeline(1)
- Do a blob detection (5)
- Track the blobs (5)
- Decide if it moves left or right (2)
- Look for potential optimization (3)

4.4 Video to Depth map (Functional, Product, Derived, Emergent requirement):

Take a video footage and convert it into depth map footage.

4.4.1 Priority

High: This is our goal for end of this sprint and the feature that our users are going to be interested in the most.

4.4.2 Estimated time

25

4.4.3 Details

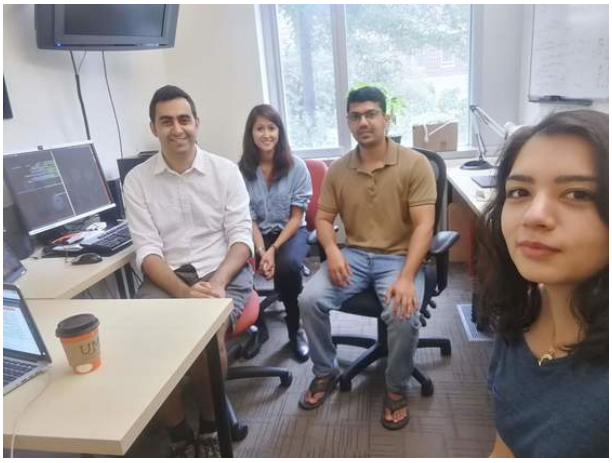
The user will be able to use any video that they have made or recorded and convert it to a depth map video footage as long as the video has either moving objects or the camera itself is moving to left or right.

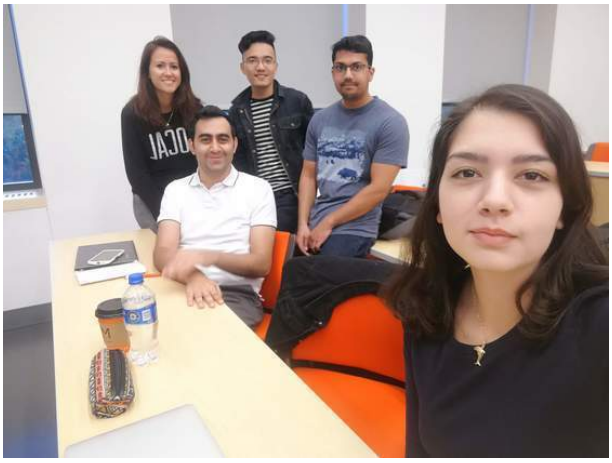
4.4.4 Tasks

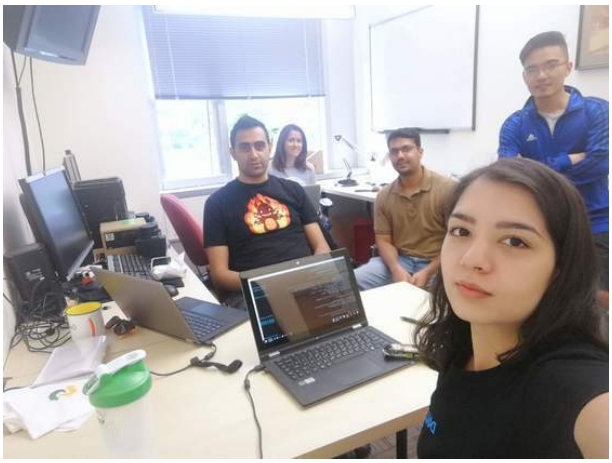
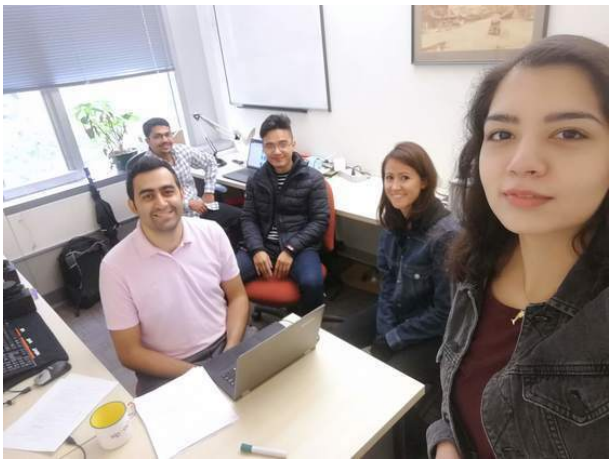
- Unit test (5)
- Setup OpenCV pipeline and take a video footage as input (1)
- Convert it to image stream (1)
- Take 2 images at a time
- Convert images to depth map (5)
- Save into output stream (5)
- Save output stream as mp4, avi or other video formats (5)
- Look for potential optimization (3)

5 Daily Scrum Photo Gallery

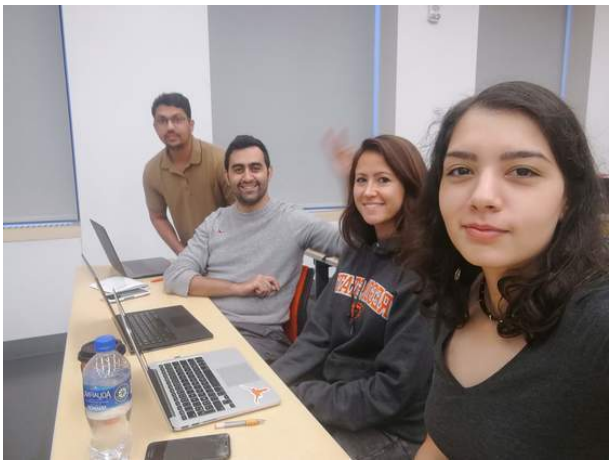
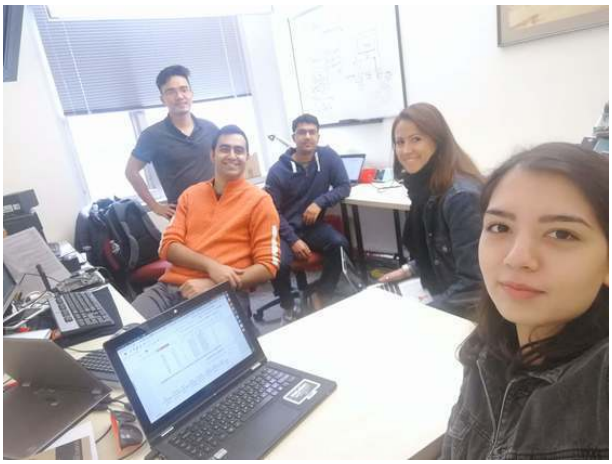
5.1 Week 1

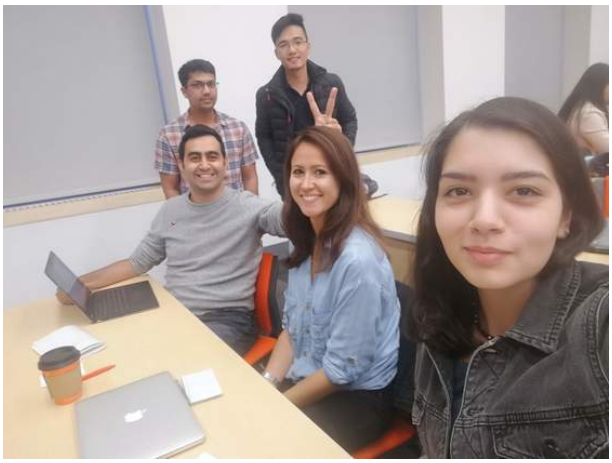


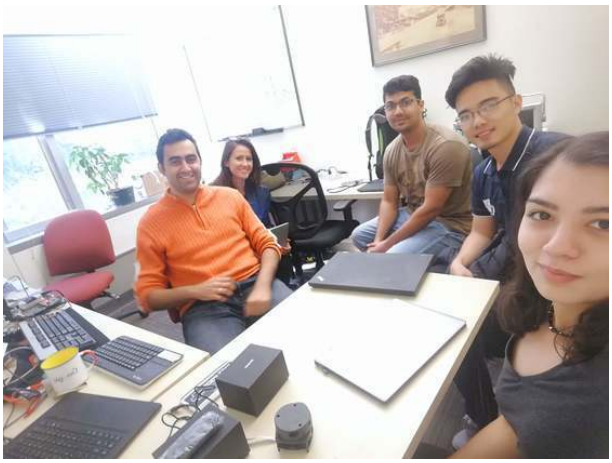


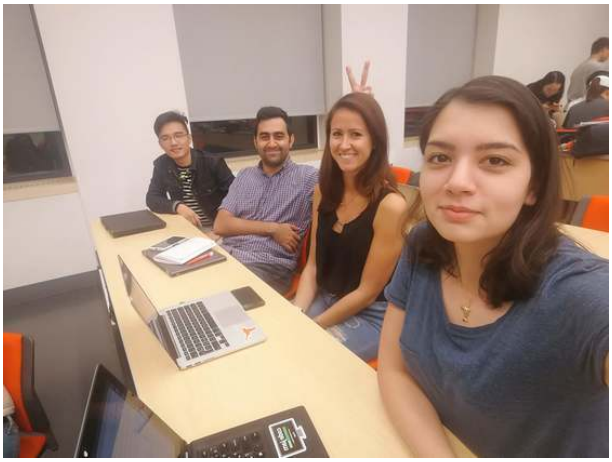
5.2 Week 2

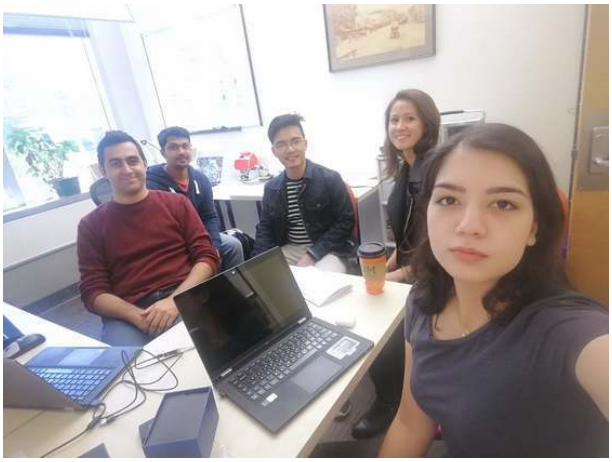



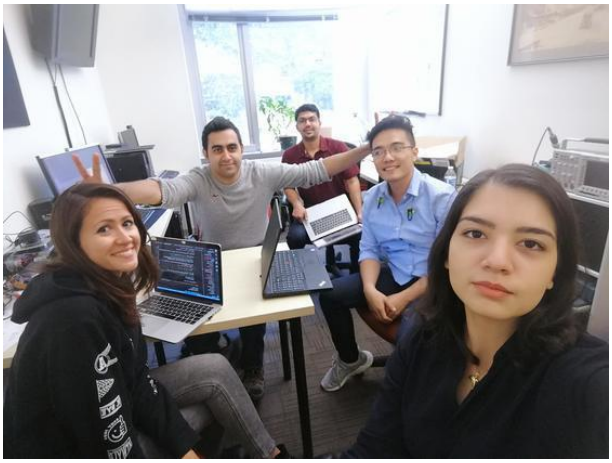


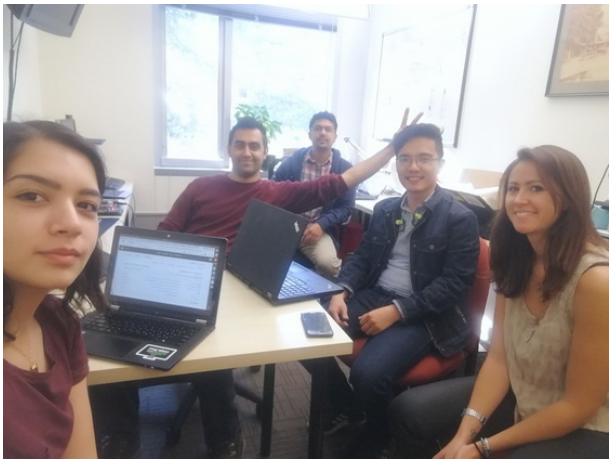


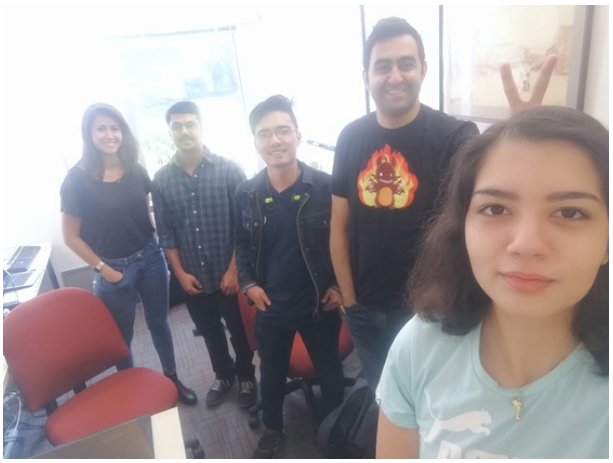
5.3 Week 3











6 GitHub Integration evidence

```
$git log --pretty=format:"%an: %s"
...
Kamilla: Added instruction how to run tests
...
BeNsAeI: Added images, files integrated, Tested
...
Kamilla:  Added new unit tests and fixed integration tests for depth map functions
...
BeNsAeI: Merge branch 'master' of https://github.com/BeNsAeI/PADSR
...
Nghi Duong: Merge branch 'master' of https://github.com/BeNsAeI/PADSR
Nghi Duong: integrate depthmap creator to convert video to depthmap video
...
BeNsAeI: test
Kamilla: Fixed unit tests and added --fast arg for video converter
...
bensaei: Integration, folder naming became more intuitive and ready for review and submission
...
Nghi Duong: Merge branch 'master' of https://github.com/BeNsAeI/PADSR
...
Nghi Duong: add timer to track
...
Margaux: Import the optical flow folder
Margaux: Merge branch 'master' of https://github.com/BeNsAeI/PADSR
...
Dhruv: feature-image-to-depth-map-code
...
BeNsAeI: Integration folder added
...
```

7 SPRINT REVIEW LOGS AND NOTES

The sprint review overall went smoothly and gave us the opportunity to discuss the outcome of our sprint in a very systematic and orderly fashion. In this section we will discuss a brief summary of what happened.

7.1 User Tests

The user for our tool was an electrical engineering masters student. In order to have a fair assessment of the setup we needed someone with a reasonable knowledge of what we are trying to achieve and some skill sets to be able to use it. Ernie Bodle was our user for this sprint. here is a brief summary of his thought on our project.

- **User story 1: Image to Depth map:**
  The first attempt did not produce the results that the user was expecting. The reason to that was mismatching resolution for the images which resulted in poor quality out put. This was mainly because the image was in portrait mode.

- **User story 2: Direction of movement in video footage:**
  User was not interested in this feature and therefore did not care enough to test it. However, he ended up using this since it is included in the next stage. The user did end up using it but not directly.

- **User story 3: Video to depth map video:**
  First attempt forked in a very satisfactory fashion. The user was very pleased with the outcome of the program and was excited enough to send the video as a SnapChat message to his friends. The video was produced very quickly and the results are very good. In our user's test he got the best results with the following command:

```
cd pad/Integration/Video_To_Depthmap
python video_converter.py -i test.mp4 -o test3.avi --low --fast
```

7.2 Product owner

Behnam Saeedi was the product owner for this sprint. He also came up with the original idea for the project. This project is based on Behnam's thesis and that helped the team to have specific requirements and reduced the time that needed to be spent to produce a project for this class.

7.3 Scrum Master

Kamilla Aslami was the scrum master for this sprint. She had the most amount of prior experience in an agile environment. Selecting her as a scrum master was a good idea since it helped us understand what are the duties of the scrum master during the course of a sprint.

7.4 Attendees

The attendees were limited to the minimum number of people necessary in order to have an effective Sprint review.

- Developers: All the members of the team were present during this meeting.
- Product owner
- Scrum master
- User

7.5 System strengths and weaknesses

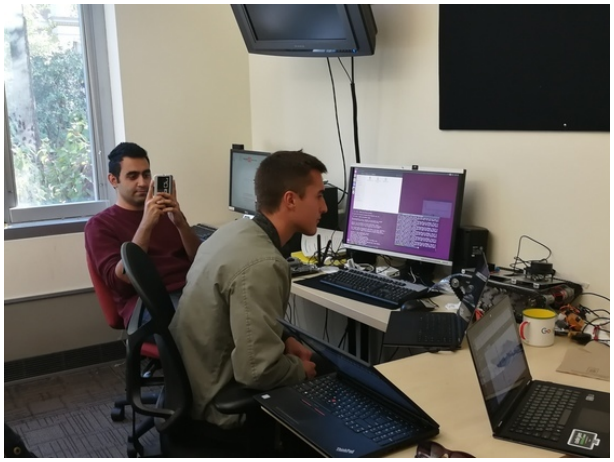
7.5.1 Strengths

- System performs fast and is very responsive
- The system is simple to use despite the fact that it is a command line user interface.
- The results are reasonable accurate
- There are a wide variety of options between performance and speed for user to chose between
- The project is usable, simple and elegant
- System is deployable on embedded systems

7.5.2 Weaknesses

- The system is not very testable due to nature of computer vision
- there are features that would behave oddly when input had unexpected form. For example bad dimensions or aspect ratios
- running time of the program is not linear and large changes in input size yield orders of magnitude increase in processing time.
- The system is very dependant on specific hardware (Nvidia GPUs) for optimal performance

7.6 Evidence



8 SPRINT RETROSPECTIVE NOTES

8.1 Notes

What went well:

- **Good estimations:** The original time estimations which we made were very realistic and we ended up achieving those times.

- **Effective split work:** The work split was fair and we tried our best to make sure no member of the team was over worked.

- **Team work - helped each other:** There were times that despite our attempts to make sure the task would be fairly distributed, there were a few tasks that surprised us by how difficult they were. In order to manage such tasks we immediately redistributed the work load. Luckily the group members were all excited about the project and were willing to take on new challenges.

- **Overall development went very well:** The team put very good effort in order to make sure the project's quality and performance were good and up to expectation.

- **Learning process went smoothly:** The spike user story was one of our major concerns. However, we can call this spike user story a success.

- **Team is excited about the project:** The team is very excited about the project and are looking forwards for next challenges that they need to face.

Things to improve:

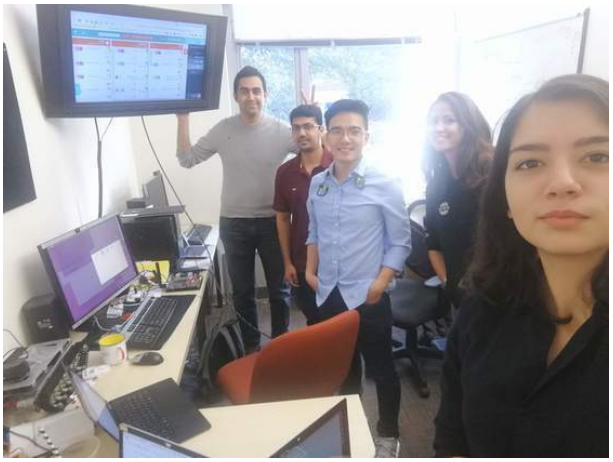
- **Integration should be started earlier:** We underestimated the integration task. We were caught slightly off guard with the integration step. In next sprint we will allocate more time and effort for the integration step.

- **Unit tests should be implemented along with the features:** This sprint many of the team members were not sure what to expect from the project. This lead to us not knowing how to implement the test with the module itself. Now that we are more comfortable with the project and more confident in our capabilities we will try to make sure that the tests and the modules are implemented at the same pace.

- **Scrum meetings time that works for everyone:** The group did manage to meet everyday. However, it was very difficult for us to come up with consistent times in every day of the week to meet. For example, on Mondays and Wednesdays we met during class, during Tuesday and Thursday we met at 3:00 PM in the afternoon and on Fridays we met 10 am in the morning. This caused lots of confusions and hassle for the team. For next sprint we want to figure out a time where it is the same time everyday that we can meet and perform our scrum meetings.

- **Project structure - earlier:** This sprint we felt like we were thrown in the project and the transition from preparing a project to actually executing it was very short. This made the project very challenging for us.

9 SPRINT RETROSPECTIVE EVIDENCE



10 CONTRIBUTION BUS

In general all members of the team contributed similarly to the project. Based on the user stories and the tasks that were necessary, we made sure that if one of the members is overworked in one task, they would have less to do on other sections of the project. Furthermore, if one task was too difficult we made sure to assign more members to make it easier on the person who was responsible for that task. Some of the tasks were long and simple task wise and some others were difficult and short.

10.1 Behnam Saeedi

My responsibility for this sprint was mainly test tools and optimization. I helped with the spike stage to help everyone setup their tools and development environments and could be found in various install scripts in different folders. Furthermore, I developed a general testing tool. The testing tool had several capabilities including memory scanning and object and class tracking through the execution. Furthermore, I was responsible for compiling the OpenCV and OpenCV Contributor's distribution with CUDA support. This process was very long and took several trial and errors but eventually it worked and helped us with our Optimization step.

10.2 Margaux Masson

Developed the camera motion detection code using OpenCV library. This code takes two frames as input and precises if the camera is moving to the right or the left, so we can know which one of the images is the right one or left one because we need to know that when we use the create depth map function. First, it was supposed to be a blob detection code which will then detects in which direction the camera is moving, but the code was too heavy and the execution was slow, so I used the optical flow to optimize this part of the code, which was then integrated into the video converter code.

10.3 Kamilla Aslami

Implemented a ”Video Converter” feature that takes video footage and converts it into a depth map footage. This user story not only consists of video processing part but also includes integration with other features. The developed functionality reads input video frame by frame, calls motion detection function to identify the camera direction, then calls a depth map function, and saves resulting depth maps to a new MP4 video.

10.4 Nghi Duong

Collaborated with Dhruv to work on creating image to depth map, and compared different methods to compare quality of output depth map images, and benchmarked each methods. I also wrote a testing script to test with different set of parameters. Integrating depth map creator function to the system.

10.5 Dhruv Jawalkar

Collaborated with Nghi on creating image to depth map using OpenCV, created CreateDepthMap class. The class takes in pictures of an object from two different view points and tries to estimate the depth based on how many pixels the object shifts. Concept is, that pixels in the background don’t change by much, nearby objects change by a lot. This phenomenon is used to estimate depth.

Also looked into the possibility of using neural nets to generate depth map of an image from a single view point.