Reducing dizziness when using a video-see-through head-mounted display

Segovia Barreales, Richard

1 INTRODUCTION

LAST summer during my internship I developed a basic video-see-through viewer for a head mounted display (HDM) prototype developed in the CVC[1], the main goal is to improve this viewer making it expandable for future modules and reduce the adverse physical reactions that it can produce to the users [18], [1]. It has to be said that this research project is focused mainly in the software, the prototypes and other hardware aspects will be out of our concerns and will be developed by others researchers of the CVC. A communication channel though is opened to discuss about the development of the whole project.

The head mounted displays first appeared in 1965 when Ivan Sutherland developed the first HDM called "the sword of Damocles" [16], it paved the ground to further development in the field. The development of this technologies have grown in the last years mainly centered in the videogames field. Some examples of this are the Oculus [9] in figure 1 or the HTC Vive [5]. These products are virtual reality headsets, therefore they are only capable of showing computer generated scenes. The developed project prototypes differ in this matter because they can also show the real world, this kind of HDMs are called video-see-through.

2 OBJECTIVES

2.1 Initial objetives

First we will be developing a software that will be capable of capture an stream from a stereo camera and display it in real time. The software will show one camera stream in each half of the screen, also the HDM will enable only to each eye to see one half of the screen, all of this will simulate the stereo effect, creating a depth effect. Another goal is to prepare this software to be easily expandable with new modules in the future.

Secondly, as is reported in [18], and we discovered in a user testing sesion in the CVC [1], using head mounted displays can cause a variety of adverse physical reactions, since this HDM will be used while working, these symptoms will reduce the concentration and the effective work-

---

• E-mail: richard.segovia@e-campus.uab.cat

• Menció en Computació

• Project supervised by: Coen Antens (CVC) and Felipe Lumbreras (Computació)

• Course 2017/18
[1] Computer Vision Center located in the UAB campus.

Fig. 1: Oculus rift head mounted display.

ing hours. For that reason reducing these adverse physical reactions is a priority. To achieve this goal, we will try to apply two techniques:

- Accommodation-Vergence: As is reported in these [18], [8] articles the mismatch between accommodation and vergence in head mounted displays causes a conflict on the expected depths increasing the feeling of discomfort and dizziness. One idea to reduce this effect is to dynamically move the position of screen frames as the focus changes from closer to distant objects and vice versa. To achieve this we will use a neuronal network that will be able to discern between closer and distant objects, also we will need the depth map obtained via the stereo camera setup.
  In addition to this, a progressive change from one focus to the other may be needed, as a big change in focus can induce sickness to the user.

- Depth of field (DoF) blur: Recent investigations [3] suggest that applying a DoF blur to a scene viewed using a head-mounted display can reduce visual discomfort, the challenge here is that our project has to make this DoF blur in real time in a real world environment, in contrast to the developed in that investigation where they used computer generated scenes. To achieve this we are going to use disparity map and the information obtained by a neural network that discerns between closer and distant objects.

Thirdly, as the HDM will be used in workplaces, the idea of adding data to the environment, could improve the workflow and the work efficiency. For this reason its think, that adding a third camera to the system can add valuable information that can be mixed with the environment. A similar idea can be found here [13] where a third camera is used to add information to the real world. For example with adding

a infrared camera, the user can be warned by highlighting objects too hot to be touched.

Fourthly, as we discovered in a user testing done in the CVC [1] last January, using a computer while wearing the HDM is a difficult task, mainly because the screen was unreadable for the brightness saturation of the image. For that reason a system capable of detecting computer screens and adapt the brightness, contrast and others parameters of the image stream will be developed.

Along all the development, user testing sessions will be taking place in order to check the improvements made between the different versions of the developed software and the different prototypes developed. The main goal of this sessions will be to evaluate if the developed software works and if it reduces the sickness feelings of the users. The procedure used in the user testing will be similar to the performed in [3], where the Simulator Sickness Questionnaire [12] where used to evaluate the symptoms.

2.2 Progress and changes

After these firsts weeks of development some objectives have change and other have appeared, as we will explain in section 4, the main issue was that the depth map generation needed some work before integrating it into the system.

The integration of a reliable depth map generation was essential to the project, it is needed to complete both Accommodation-Vergence and DoF goals, and as it give us knowledge of the enviaroment it would probably be key in future developments. For these reasons it was decided to add it as a objective itself. The main goal will be to integrate a fast and reliable depth map generator with all of the required components.

As the depth map generation requires a stereo pair of undistorted images in order to obtain a reliable depth map, the integration of a system capable of read calibration files or calibrate itself and undistort the output of the cameras is required.

The performance of the depth map generation is also key, as a slow image generation will give the system great latency and that could limit the uses cases of the system.

The current objectives tree can be found in the figure 2 in the appendix, it was done to recollect all the goals that this project want to achieve. As we will see in 4, not all of this goals will be reachable because of time constraints.

3 METHODOLOGY

Scrum and its variants are one of the most spread work methodologies nowadays. The scrum methodology is the selected to be used in this project, however, as this project will only be done by one person, some changes have to be made.

First as there is only one developer the daily meeting will be substituted with a weekly meeting with the stakeholders, in this case the tutor and the boss of the laboratory department in the CVC, in these meetings we will evaluate the development done, the issues faced that week and the problems solved. In addition to that we will discuss and review future milestones and the progress towards them.

Secondly, a backlog will be prepared at the beginning of the project and will contain the main goals divided in tasks, these tasks will be organized in groups of sprints easily done in a week. As each sprint has a backlog of task to be done, a tool can be used to organize this backlog, in this case Trello[2] will be used.

Thirdly, each iteration over the documentation and the development will be kept by a version tracker, in this case, github [10] and its desktop client [11].

Keeping the main scrum methodology, I think it's interesting to grab some ideas from the Lean software development [6]. The idea of removing the so called "Muda", non important extra features and processes, could be beneficial for this project because it could reach an overwhelming dimension for the limited time that is have.

Involving the tools that will need to be used develop the project, first of all, the current viewer version is made in C++ using the QT libraries [4], therefore any tools that would be used need to be compatible with the current version of the viewer. For that reason, further investigation will be performed to identify which neural network library is more suitable.

The custom scrum methodology already explained has been followed without any change in the first weeks of development. Every part of the development process has been documented and every change in the code has been uploaded to a private github repository. It is worth mention that the OpenCV [17] library has been used to develop the calibration and undistort phase.

4 PLANNING

4.1 Dataset

To be able to develop this project a dataset will be necessary to train the neural networks and to check the depth map. In this case as we have available three hardware prototypes, a data collection procedure will be possible and simple. For that reason a dataset caption session is planned.

Also a dataset will be used to make sure our dataset is correct and to encompass a broader range of environments. For that reason we selected first the Middlebury stereo dataset [15] because it has a great dataset of stereo images and the groundtruth of their depth map. Additionally a dataset with indoor and outdoor scenes is recommended. Several dataset could be used with that purpose, for example the MIT's places dataset [19] and the INDSECS indoors only dataset [14].

Also a calibration dataset is required in order to generate a quality calibration. This dataset will contain a series of images of a chesslike pattern [?] located in differents parts and depths of the image. Is required that the patterns appears completely in both camera. This dataset will be updated every time the cameras or the HDM gets manipulated, this is because any minor change in the focal length or the position between the cameras will affect the results of the calibration.

4.2 Accommodation-Vergence

In this stage, a neural network will be trained to detect the distance of the focus and the necessary vergence in the

screen placement to help the user a more comfortable experience. This will, alongside with the DoF blur, be the main time-consuming stage of the project and will have the following parts:

1. Preset creation: In this task, the current viewer will be expanded with the creation of multiple and easy to change presets that will allow further user testing.

2. User testing, presets: In this first user testing session after the finalization of task 1, the users will be told to configure their own focus settings and to try to change between them to adapt the focus to closer or distant objects. The goal here is to acknowledge whether the user feels better accommodation with the focus between objects.

3. Progressive preset change: After the development of 1 and 2, if the users report sickness because of the sudden change between presets, a smooth transition between them will be developed to reduce the issue.

4. Disparity map: In parallel with the development of the tasks 1, 2 and 3, a disparity map generator from a stereo pair is required to continue with further development. As this subject is complex per se and is not the main goal of this project, a library that already implements a good stereo matcher, like LIBELAS[7], will be used.

5. Neural network with camera streams: This task will start after the analysis of the data collected in the task 2, an the main goal will be to create a neural network capable of returning the value of the vergence. To achieve this several architectures will be tested and compared, with one, two or three streams from the cameras and including the disparity map information when developed.

6. User testing: In this user session, the goal will be to check if the development done in this stage has reduced the sickness feelings of the users, and with which solution the user feel more comfortable.

4.3 Depth of field blur

This stage will be developed after the finishing of the previous Accommodation-Vergence phase 4.2 and shares some development with it.

This problem can be resolved in to ways, using a disparity map only to create the DoF and using Neural network to aid the creation of it.

1. Mock up blur: to ensure that the results of [3] are applicable to real world scenarios, a mock up blur photo dataset will be made, and a user testing session will take place.

2. Designing de pipeline: this task goal is to design, both pipelines, the disparity map only and the neural network aided.

3. Disparity map version: this version will generate the DoF blur using only the disparity map to get the depth and using the center point of the cameras as the focus point.

4. Neural network adaptation: this task will take the neural network developed in the previous phase and will modify it to make it usable in this phase of the project. The idea is that this network will produce an indicative value of the blurriness needed using the camera images to determine whether the scene has closer or distant objects.

5. User testing: Along with the development of 3 and 4, a user testing session will take place to check if this system has reduced the dizziness and which solution is more comfortable for the user.

4.4 Improving screen quality

In this stage a system to improve the quality when the user is looking a screen will be developed. This system can be split in two tasks:

1. Quality improvements: In this task a system capable of changing contrast, brightness and other image parameters will be developed, this system will not be automatic and will need the interaction of the user.

2. Automatic switching: The goal of this task will be to free the user from the interaction with the system developed in task 1, to achieve this a neural network will be developed to detect when a user is looking a screen.

Part of the development of this objective will be left to future projects because of time constrains.

4.5 Third camera

This stage will have as a main goal the integration of a third camera in the viewer software and highlighting relevant information obtained using the third camera on stream viewed by the user. It can be split in the following tasks.

1. Integration with the viewer: Introducing a third camera in the system will mean making some modifications in part of the viewer code. The goal of this task will be to prepare the different modules of the viewer for the introduction of the third camera.

2. Research mixture: After the development of task 1, a system to allow the mixture of images between visible and the third camera spectrum will be developed, the goal will be to add information to the user from the third camera different spectrum.

Part of the development of this objective will be left to future projects because of time constrains.

4.6 Research, documentation and planning

This stage of the project can be split in three parts:

1. Initial documentation: the documentation process will start at the beginning of the project and it includes, recollect information about the state of the art, do the planning and search about the different technologies involved in this project.

2. Progress documentation: this will be written along all the project and will consist in document every change and issue that can happen. When a part of the project changes its orientation or a milestone is achieve the documentation will be updated with the changes.

3. Closure documentation: When the project will approach the finish, an article, a recompilation of all the work done and a presentation will be made to wrap up the project.

The general planning timeline can be found in the Gantt diagram, figure 3, in the appendix.

5 DEVELOPMENT PROGRESS

At the start, the firsts tasks were the development of the preset system and its user testing session. They were delayed because the main HDM prototype was being revised. Therefore the first task started was the development of the depth map system. The LIBELAS library was downloaded and some of the headers were updated to make it compatible with the compiler used, Microsoft Visual c++ compiler v15. After the firsts stand-alone test we realised that a calibration and undistort module was necesary, as the LIBELAS requires a totally undistorted image with completely rectified epipolar lines[2].

The implementation of the calibration and undistort parts was difficult mainly due to the lack and the outdated documentation of the calibration library in OpenCV C++. Finally the module was implemented as two classes, the monocular calibration class and the stereo calibration class, the stereo one uses two monocular objects to store the non-shared parameters. Both can save and load their parameters from YAML files.

Once the calibration and undistort modules were integrated, the depth map module was finally tested within the interface. The first thing noticed was the huge latency between each frame, as can be seen in the results in this is not desired because we need a high speed system that adapts quickly the vergence distance. There are two possible ways to solve the issue.

- Reduction of the image resolution: In our use case, only the center of the images is necesary, this is because it is where the user will be usually looking at, therefore taking off the outer pixels of the images will not alter the behavior of the vergence system and will improve the speed of the generation of depth map images.

- Faster libraries and implementation: Another option would be using faster or optimized libraries to be able to speedup the performance in the generation of images. As the we are using libraries already implemented our possibilities are limited in this matter, an option would be using a multithreading version of the LIBELAS library, already implemented in [7].

In this case, the first option will be selected, but if needed the second solution can also be used. In the table 1 can be

---

[2]In stereo systems, when two points that can be seen in both images and are in the same Y coordinate

---

Seconds b/w frames	Fps	Width	Height
1.98	0.50	1600	1200
1.21	0.82	1100	1100
0.46	2.16	800	600
0.21	4.67	600	400
0.33	2.97	600	600

Table 1: Frames per second (Fps) for the generation of the depth map for each image resolution

seen the current performance of the depth map generator. As well as the times, in the figure **??** we can see the full resolution disparity map of the image **??**.

Its worth mention the extremely high accuracy needed in the calibration phase in order to obtain a correct depth map, any minor change in the camera setup, can give inaccurate depth maps.

6 REPLANNING

During the firsts weeks of work some problems have arisen, the main issue was that in order to integrate a system capable of generating depth map image, is was required to undistort the image pair captured. This is because each optical lens of each camera introduce a distortion on the image captured. In addition, as the lenses are not exactly equal and they are not exactly a alienated, a stereo calibration system was required in order to produce the undistortion. This stage of the project was split in three parts.

- Calibration development: To develop the calibration and undistortion, the OpenCV library [17] was used as it has some calibration functions already implemented and is a well known library.

- Depth map development: As it is explained in 4.2, LIBELAS library was selected. One of the main reasons for this selection was the performance. !!!!!!!!!!!!!!!!!!

- Calibration dataset: Once the calibration and the depth generation systems were integrated, the calibration dataset was captured, following the explained in 4.1. This dataset needs to be updated every time the headset gets manipulated to ensure the calibration, and therefore the undistortion is accurate.

7 ACKNOWLEDGMENT

REFERENCES

[1] Coen Antens. Unpublished: User testing resume. 2018.

[2] Atlassian. Trello. https://trello.com, 2018. Last access March 9 2018.

[3] Kieran Carnegie and Taehyun Rhee. Reducing visual discomfort with hmds using dynamic depth of field. *IEEE computer graphics and applications*, 35(5):34–41, 2015.

[4] The QT Company. Qt. `https://www.qt.io/`, 2018. Last access March 9 2018.

[5] HTC Corporation. Vive. `https://www.vive.com/us/`, 2018. Last access March 10 2018.

[6] Anderson David J. Lean software development. `https://msdn.microsoft.com/en-us/library/hh533841(v=vs.120).aspx`, November 2011, updated in November 2012. Last access March 9 2018.

[7] Andreas Geiger, Martin Roser, and Raquel Urtasun. Efficient large-scale stereo matching. In *Asian Conference on Computer Vision (ACCV)*, 2010.

[8] David M Hoffman, Ahna R Girshick, Kurt Akeley, and Martin S Banks. Vergence–accommodation conflicts hinder visual performance and cause visual fatigue. *Journal of vision*, 8(3):33–33, 2008.

[9] Facebook Inc. Oculus rift. `https://www.oculus.com/`, 2018. Last access March 10 2018.

[10] GitHub Inc. Github. `https://github.com/`, 2018. Last access March 9 2018.

[11] GitHub Inc. Github desktop. `https://desktop.github.com/`, 2018. Last access March 9 2018.

[12] Robert S Kennedy, Norman E Lane, Kevin S Berbaum, and Michael G Lilienthal. Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *The international journal of aviation psychology*, 3(3):203–220, 1993.

[13] Jason Orlosky, Peter Kim, Kiyoshi Kiyokawa, Tomohiro Mashita, Photchara Ratsamee, Yuki Uranishi, and Haruo Takemura. Vismerge: Light adaptive vision augmentation via spectral and temporal fusion of nonvisible light. In *Mixed and Augmented Reality (ISMAR), 2017 IEEE International Symposium on*, pages 22–31. IEEE, 2017.

[14] Andrzej Pronobis and Barbara Caputo. The kth-indecs database. *Kungliga Tekniska Hoegskolan, CVAP, Tech. Rep. CVAP297*, 2005.

[15] Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *German Conference on Pattern Recognition*, pages 31–42. Springer, 2014.

[16] Ivan E Sutherland. A head-mounted three dimensional display. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, pages 757–764. ACM, 1968.

[17] Opencv team. Opencv library. `https://opencv.org/`, 2018. Last access March 10 2018.

```
HDM
├── Modular HDM Viewer
│   ├── Preset Shortcuts
│   └── Better program configuration
├── Accommodation-Vergence
│   ├── Preset creation
│   ├── Neural network
│   ├── Disparity map
│   └── User testing
├── Depth of field blur
│   ├── Mock-up blur
│   ├── Disparity map Version
│   ├── Neural network Version
│   └── User testing
├── Third eye
└── Screen quality
```
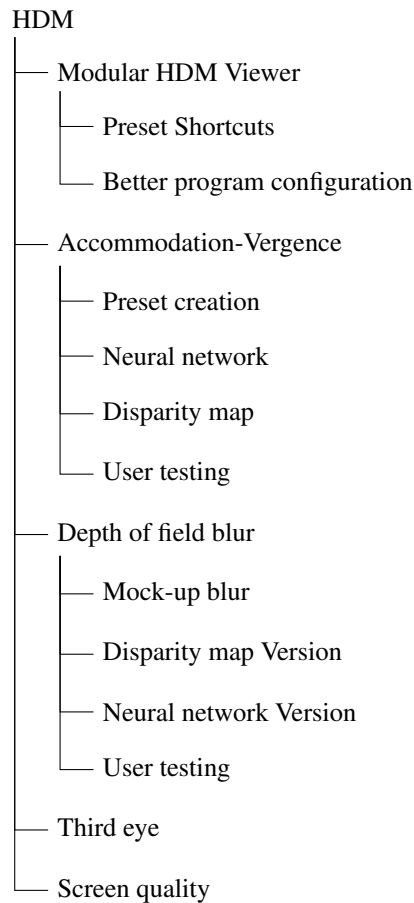
Fig. 2: Objective Tree of the project

[18] Kasim Terzic and Miles Hansard. Causes of discomfort in stereoscopic content: a review. *arXiv preprint arXiv:1703.04574*, 2017.

[19] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014.
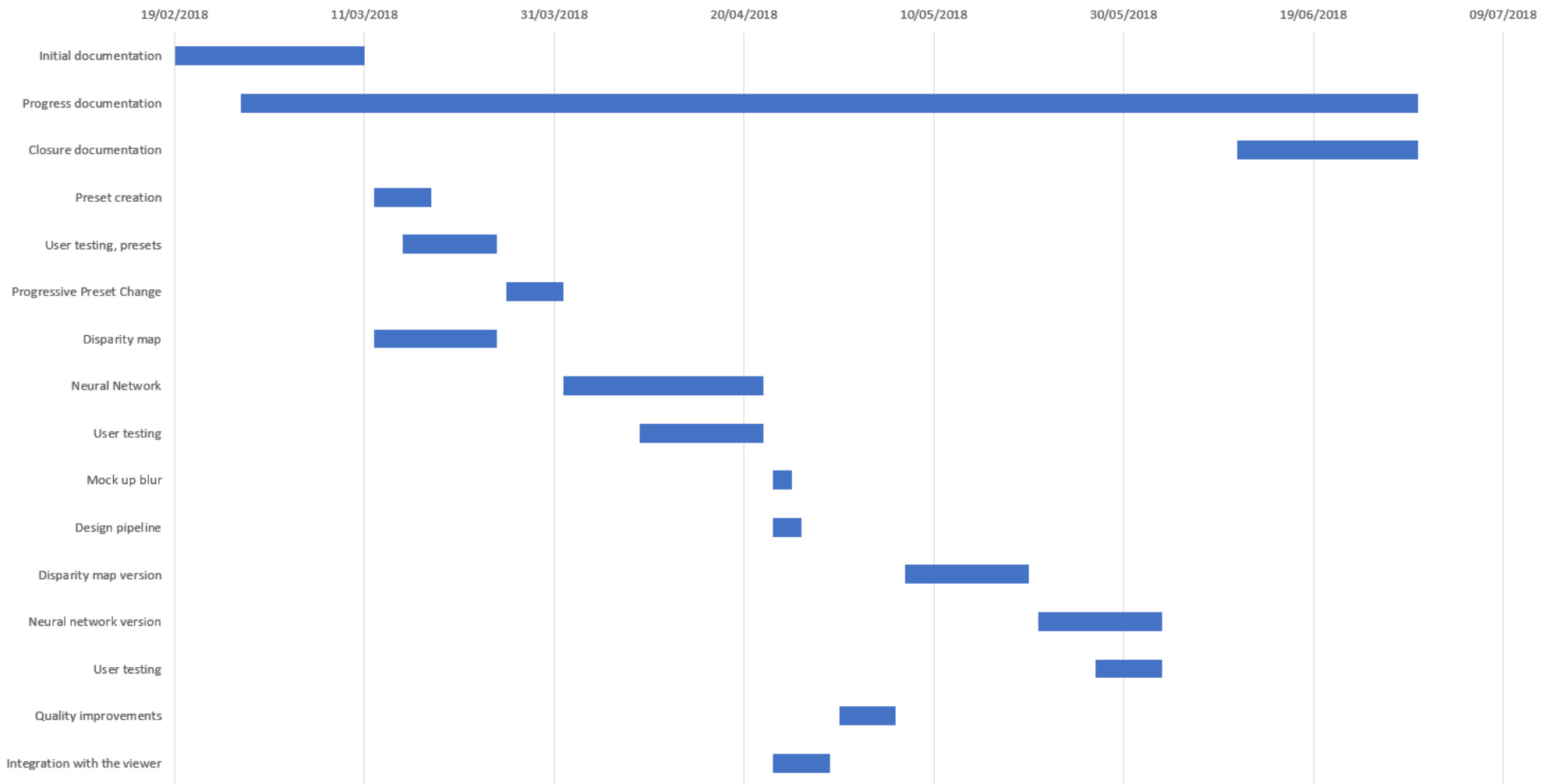
APPENDIX

Fig. 3: Gantt diagram of the project.