



Project Management and Software Development

Dyna-Eye: A dynamic 2D-3D Stereo Viewer

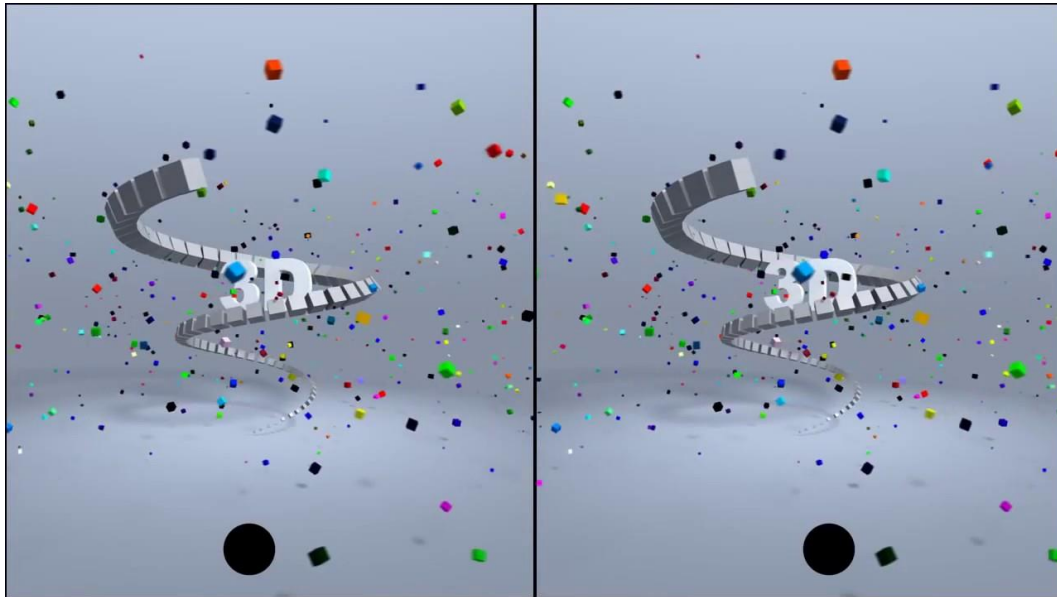
Final Presentation – 08.02.2018

Project Supervisor: Benjamin Busam

Presented by: Ruiqi Gong

Recap

- Stereo viewer with crossed eyes



[YouTube](#). "3D without glasses, Cross-Eye HD" (07.11.2017)

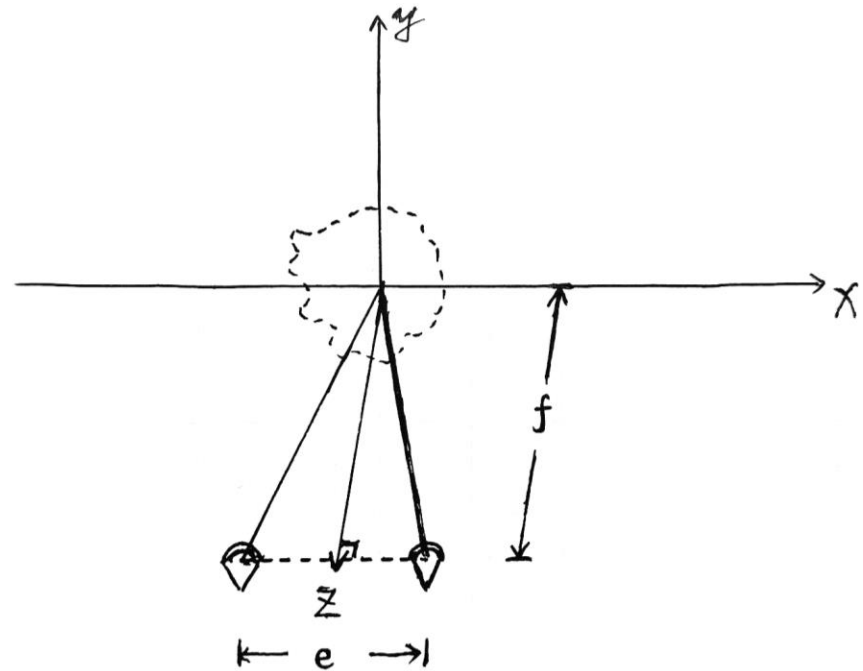


[Donald Simanek](#). "How to view 3D without glasses". (05.11.2017)



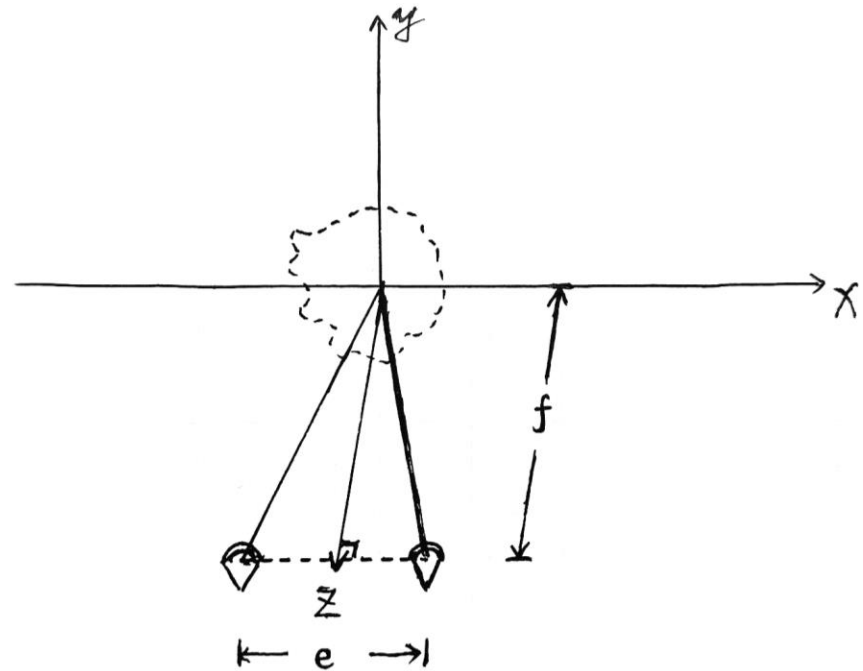
Method: local model

- Notation:
 - e : distance of two eyes
 - f : distance to focus point
- Preprocess:
 - Calculate centroid of the 3D model
 - Translate the object onto the origin
 - Normalize volume of the object



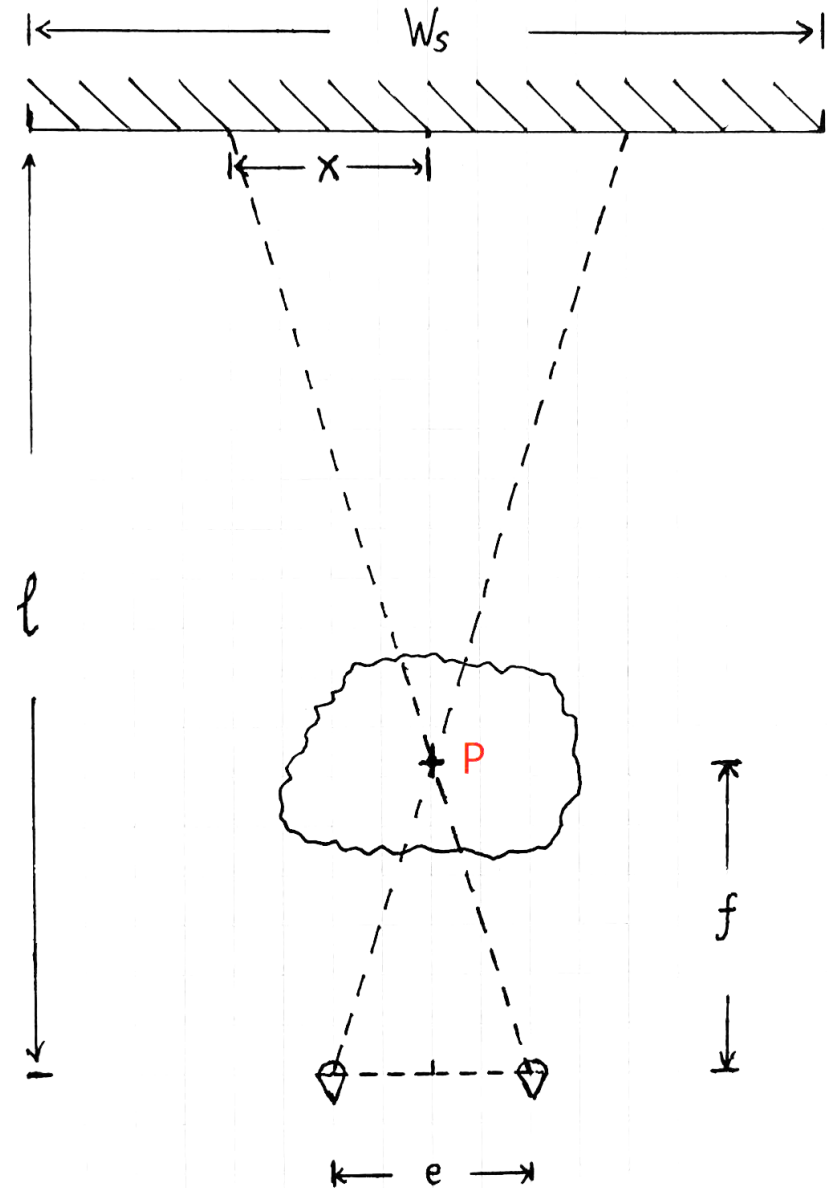
Method: local model

- Notation:
 - e : distance of two eyes
 - f : distance to focus point
- Preprocess:
 - Calculate centroid of the 3D model
 - Translate the object onto the origin
 - Normalize volume of the object
- Click! Click!
 - Take two pictures



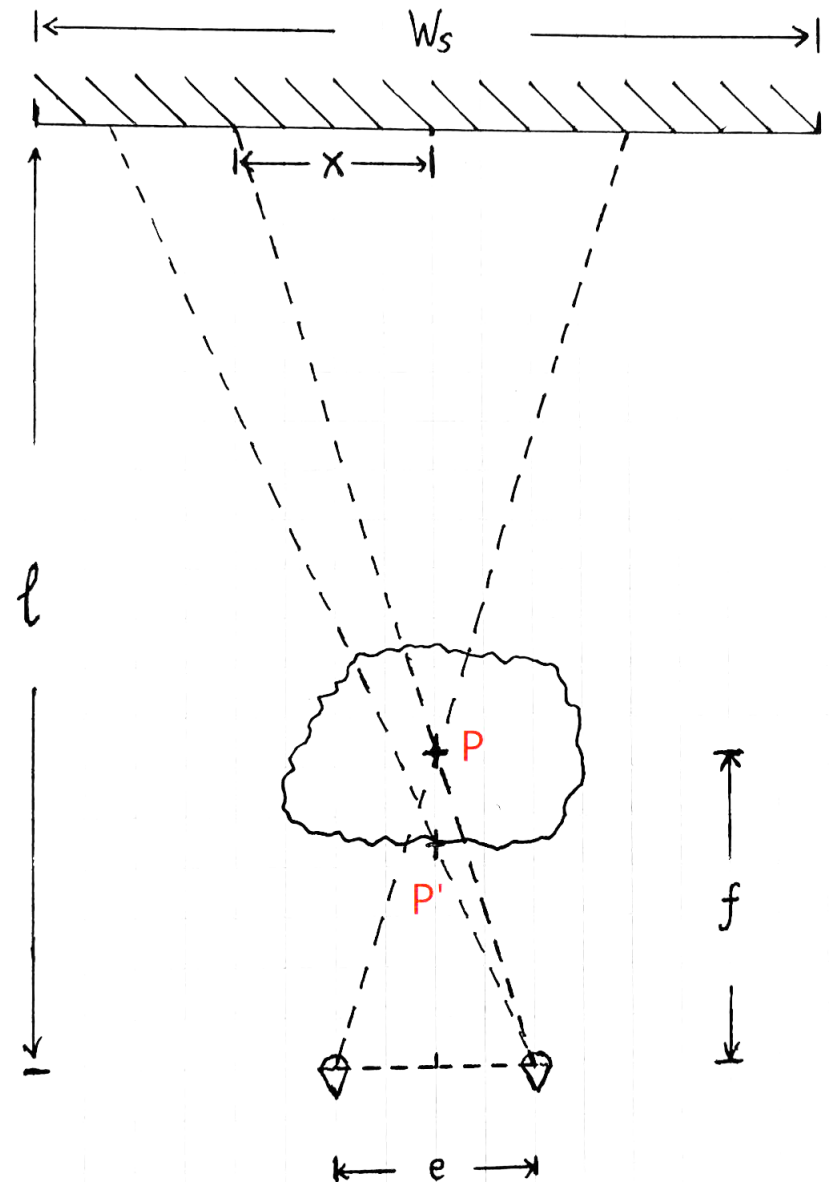
Method: global model

- Notation:
 - l : distance to screen
 - W_s : width of screen (in centimeters)
 - x : 3D model's position in viewport
- Split the window into two viewports
- Display “two pictures” on correct positions



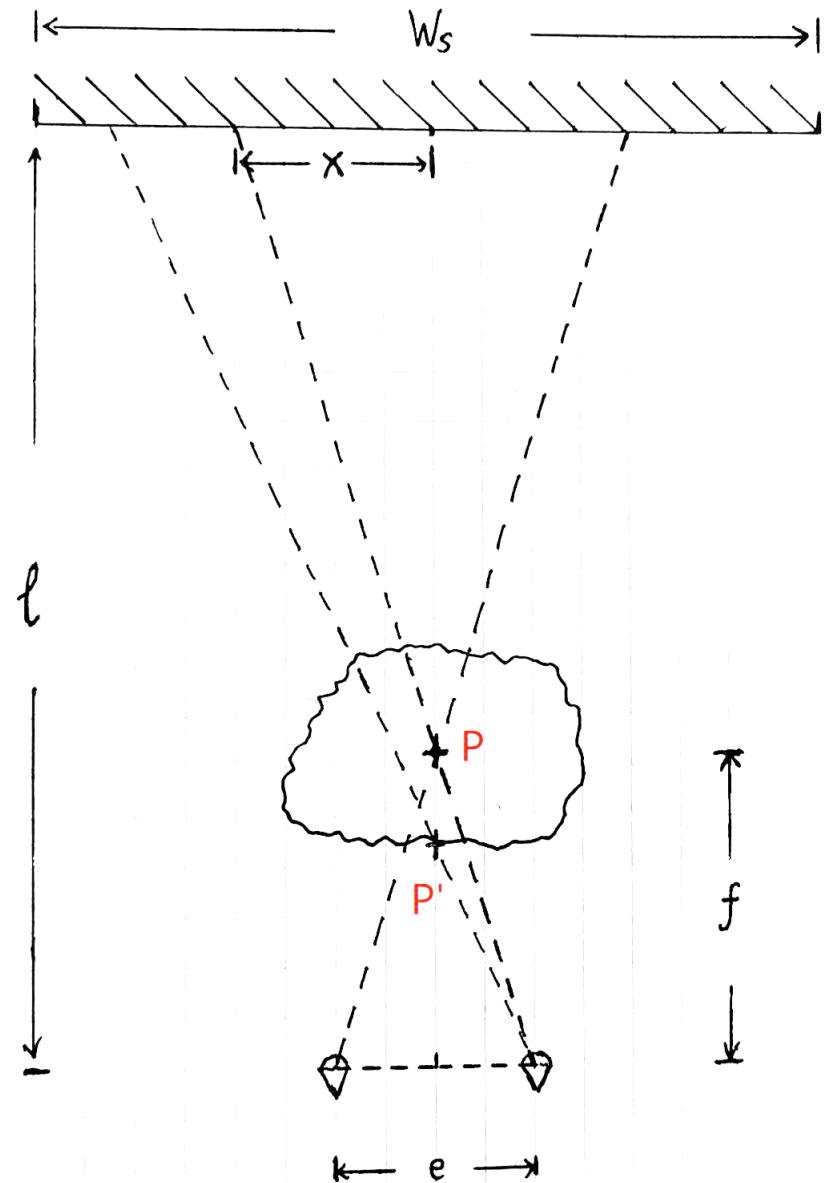
Method: global model

- Notation:
 - l : distance to screen
 - W_s : width of screen (in centimeters)
 - x : 3D model's position in viewport
- Split the window into two viewports
- Display “two pictures” on correct positions
- Shift “two pictures” according to the distance to focus point



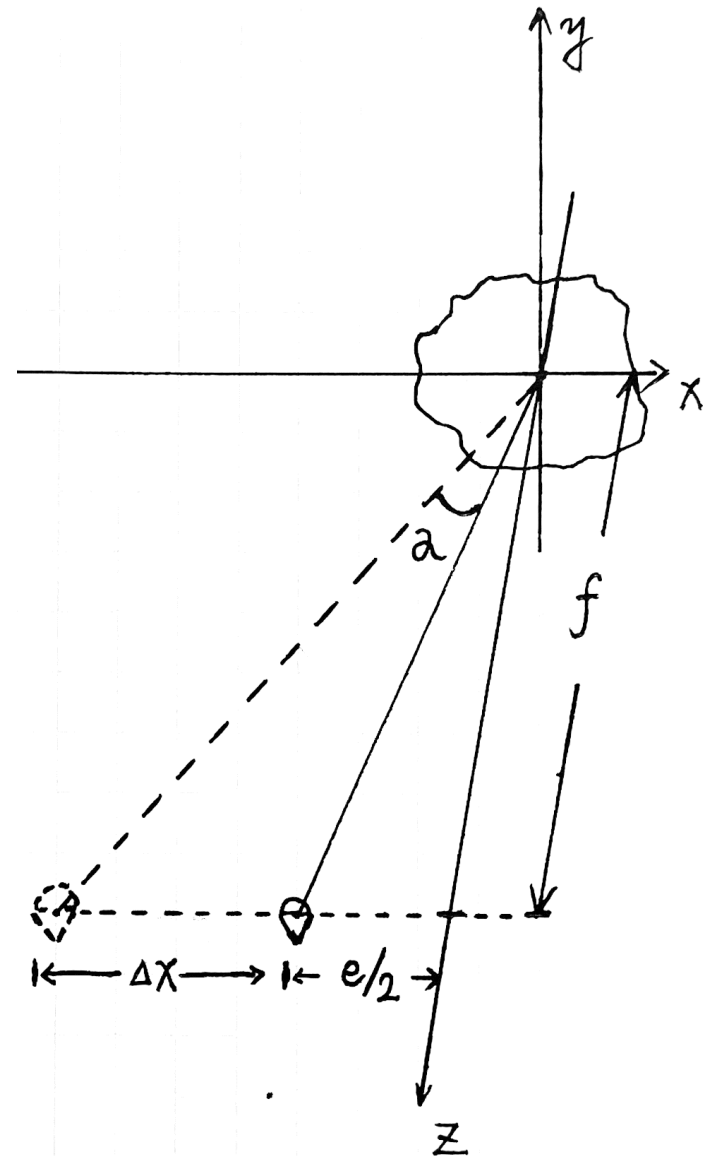
Method: global model

- Notation:
 - l : distance to screen
 - W_s : width of screen (in centimeters)
 - x : 3D model's position in viewport
- Split the window into two viewports
- Display “two pictures” on correct positions
- Shift “two pictures” according to the distance to focus point
 - Problematic: no “picture-shifting” function
 - Solution: achieved by shifting the cameras

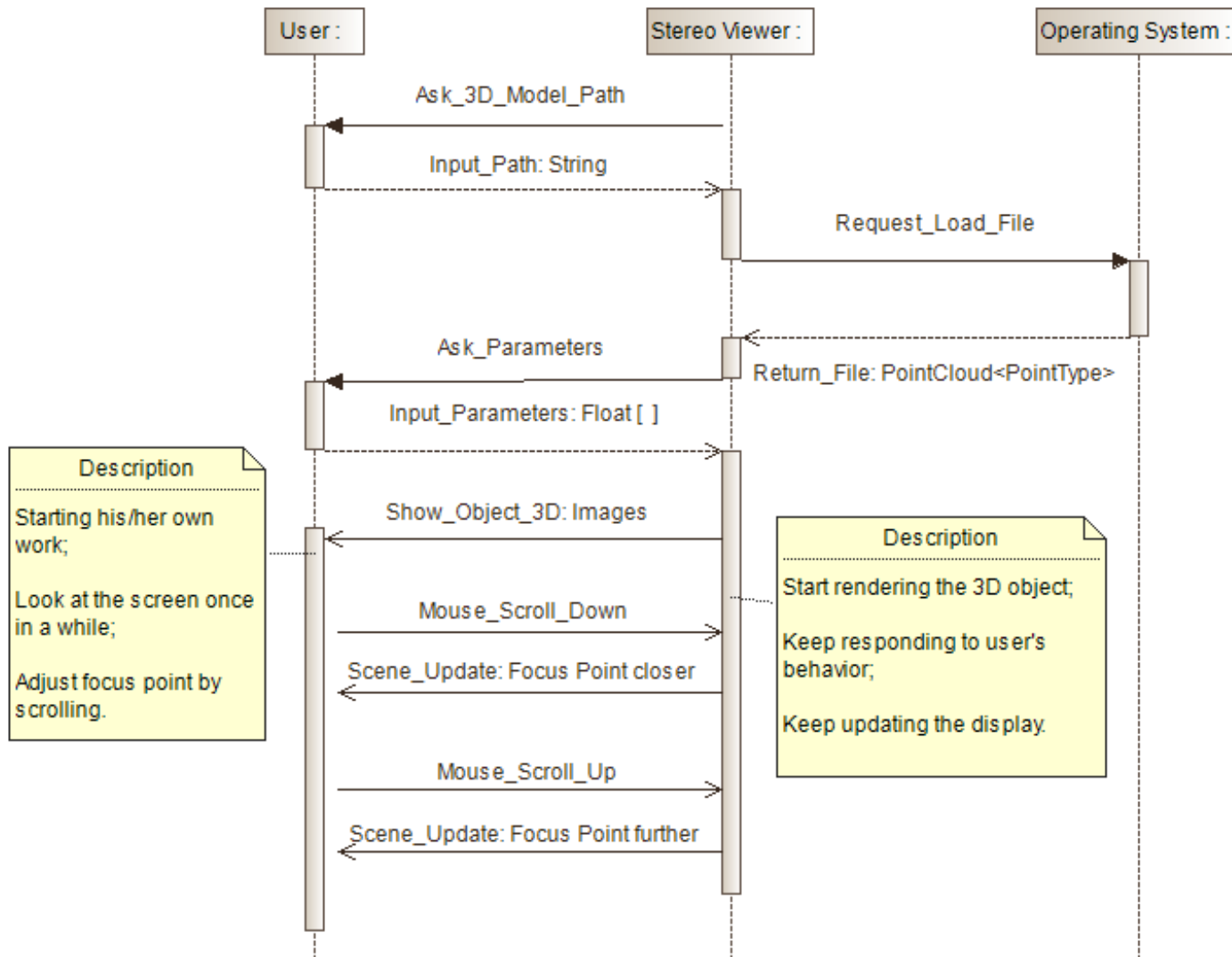


Method: rotation compensation

- Notation:
 - Δx : horizontal shift
 - α : angle of rotation
- Model is in 3D, and its image changes when the camera moves horizontally
- Rotate the model and let it face to the camera as before



UML Diagram

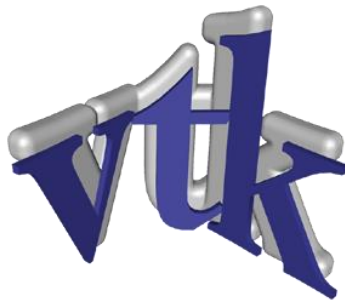


Achievement

- ✓ Gain knowledge of PCL, VTK and Cmake
- ✓ Build up the math model
- ✓ Implement the rendering pipeline from 3D model to binocular images
- ✓ Choose and create appropriate 3D models
- ✓ Implement the focus helper
- ✓ Correlate the focus helper with the 3D model
- ✓✗ Runtime focus point movement (preliminarily completed)



[PCL](#)



[VTK](#)



[CMake](#)

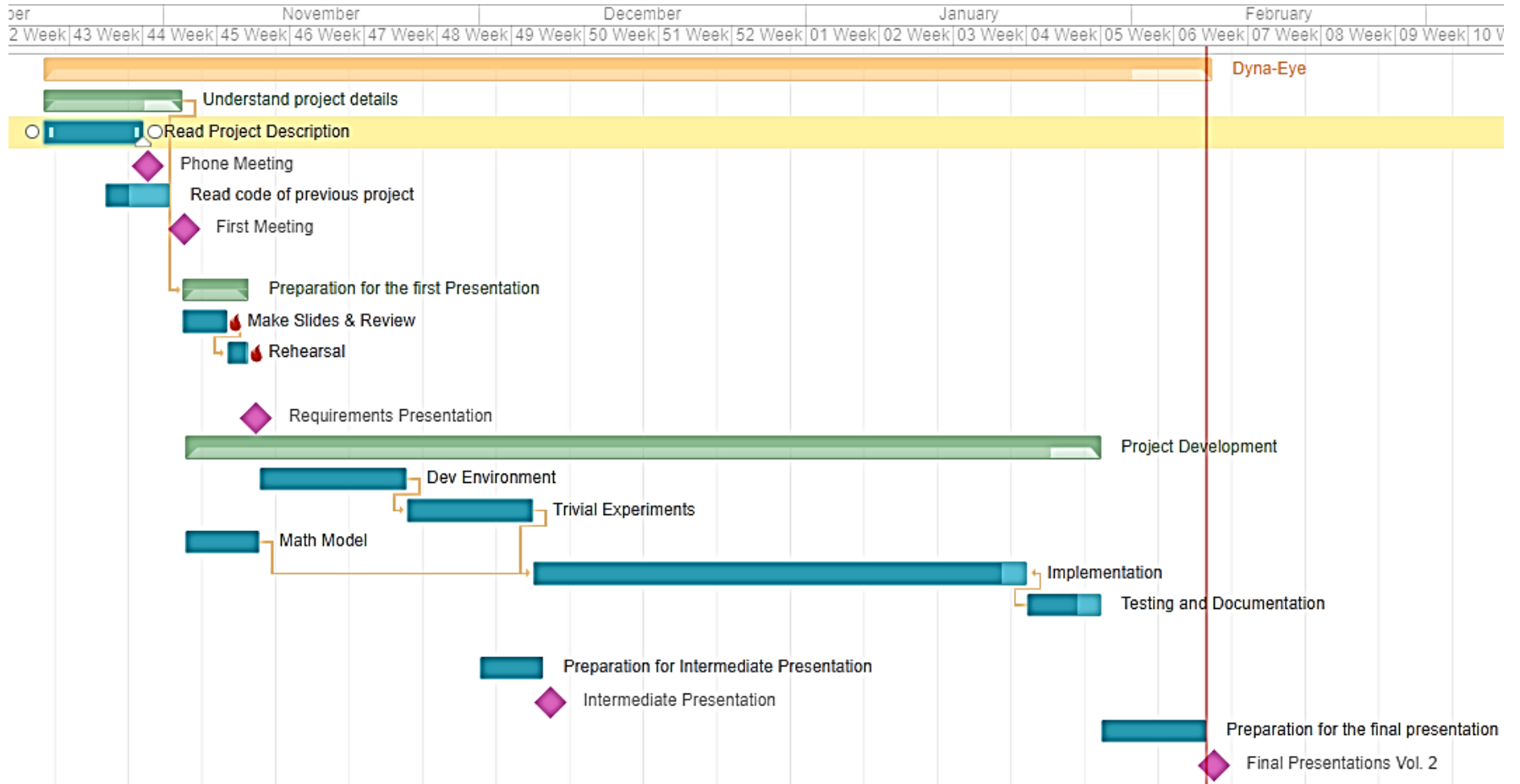


Future work

- Faster and smoother runtime focus point movement
- Better calibration between PCL coordinates system and real-world metrics
- Further testing on screens with different sizes
- Complete documentation



Gantt Chart





Thank you for your attention.

Question?