Spring 2015,CIS II Project #4

Project Checkpoint Presentation

Han Xiao
hxiao9@jhu.edu

Mentors
Dr. Nassir Navab (nassir.navab@jhu.edu)
Bernhard Fuerst (be.fuerst@jhu.edu)
Javad Fotouhi (fotouhi@jhu.edu)

1. Project Overview

Our goal is to integrate a depth sensor (Kinect sensor) into the CamC (Camera Augmented Mobile C-arm) system.

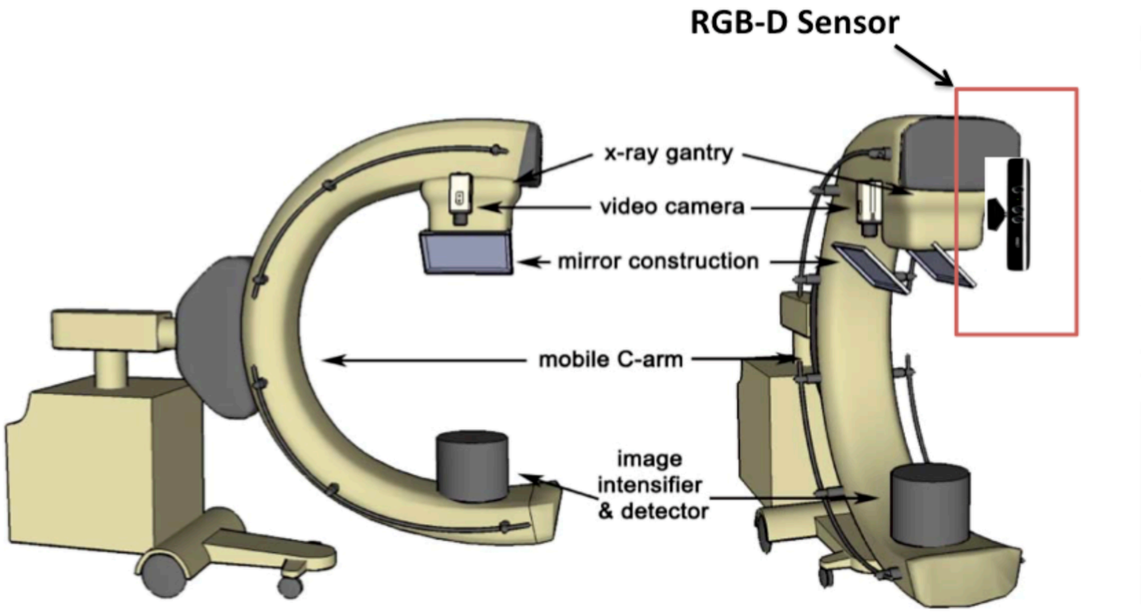
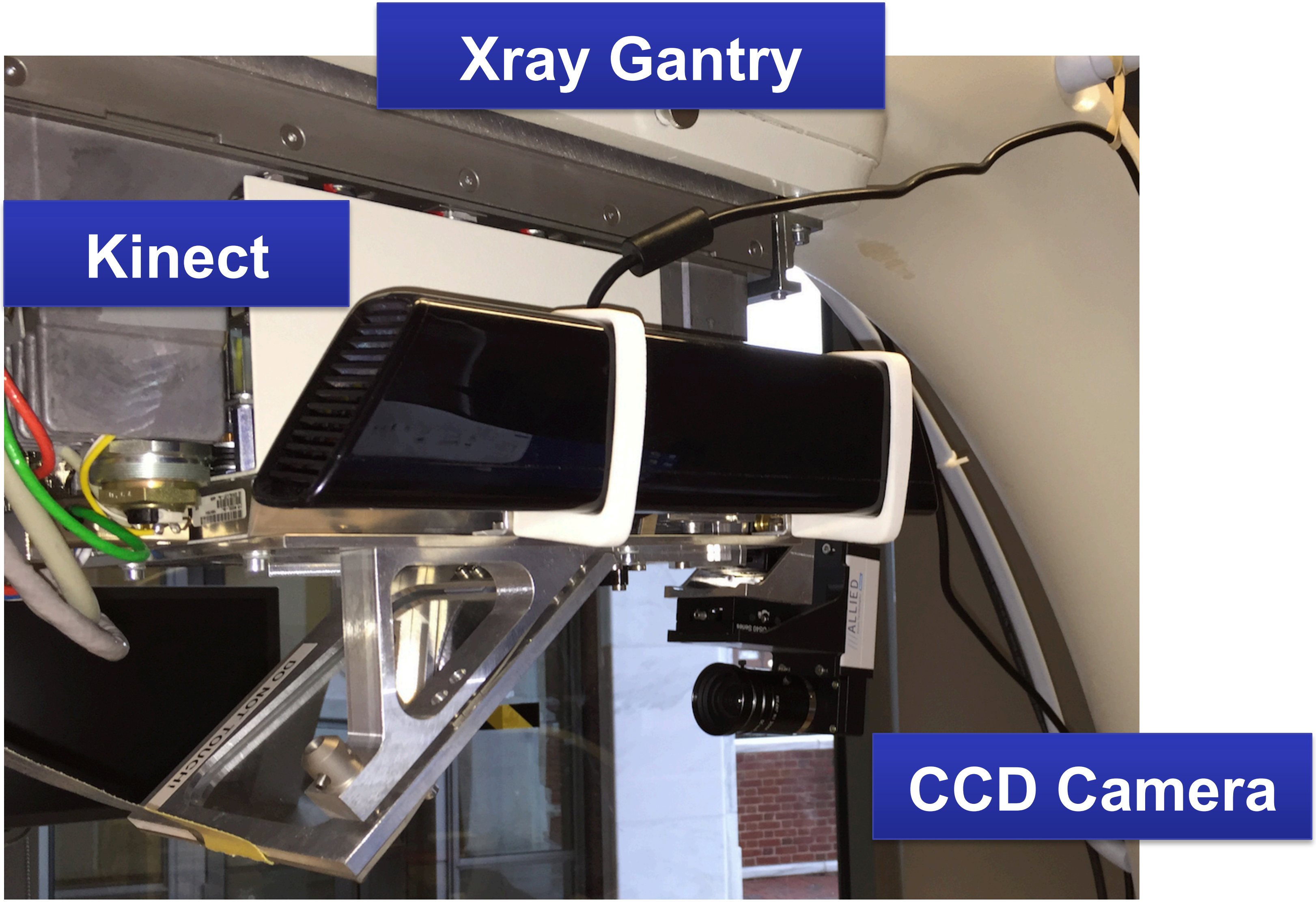


Figure 2. Illustration for Kinect mounting. (Navab, Nassir, IEEE Transactions 2010)


Xray Gantry
Kinect
CCD Camera
DO NOT TOUCH!
ALLIED

- **Hands and tools segmentation**
- **Spatial relationships determination**
- **Enhanced X-ray overlay without blocking**

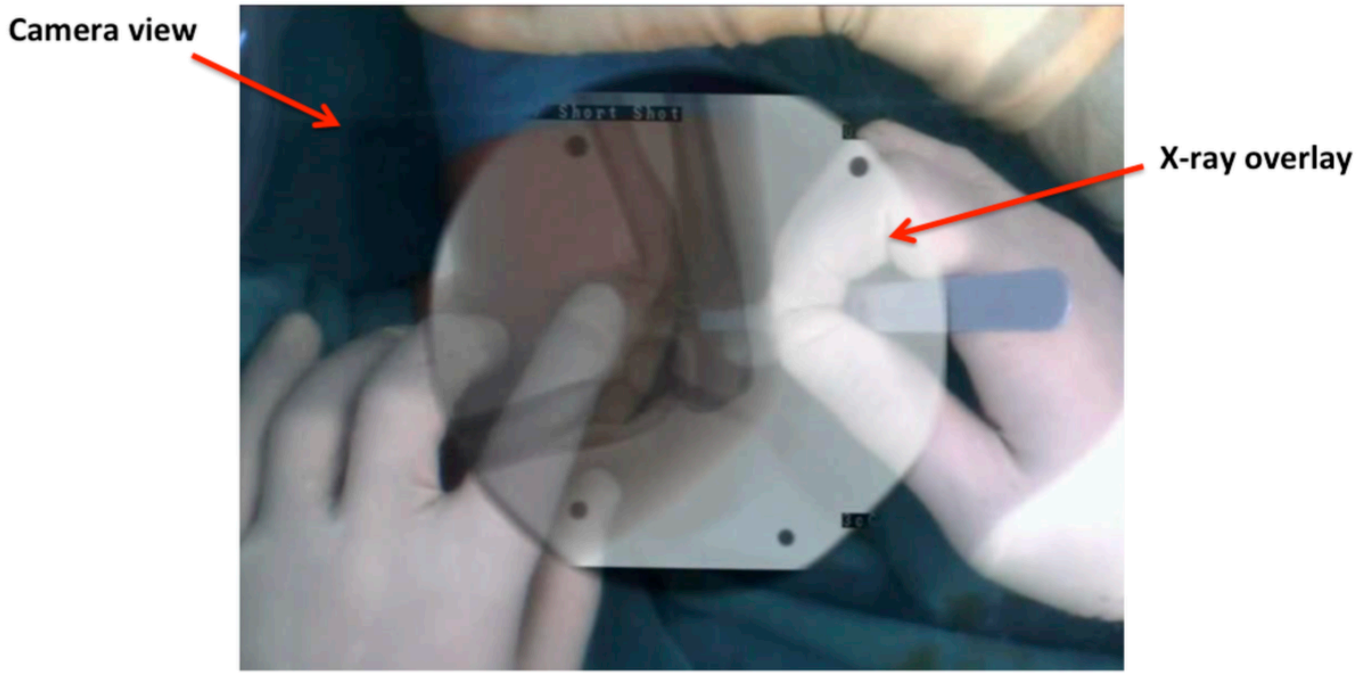


Figure 1. Overlay view of CamC. (Navab et al. IEEE TMI 2010)

2. Current Progress

4.1 Minimum deliverables

- ImFusion plugin for X-ray image acquisition, and CCD camera video acquisition. completed
- Kinect sensor mounting and point cloud acquisition. completed
- X-ray image – video calibration, and video – point cloud registration. 80% completed. X-ray and CCD calibration will be finished this week

4.2 Expected deliverables

- Enhanced X-ray overlay rendering. In progress

4.3 Maximum deliverables

- Phantom validation and surgical procedure evaluation
- Add more useful overlays according to depth information

3. Software Architecture

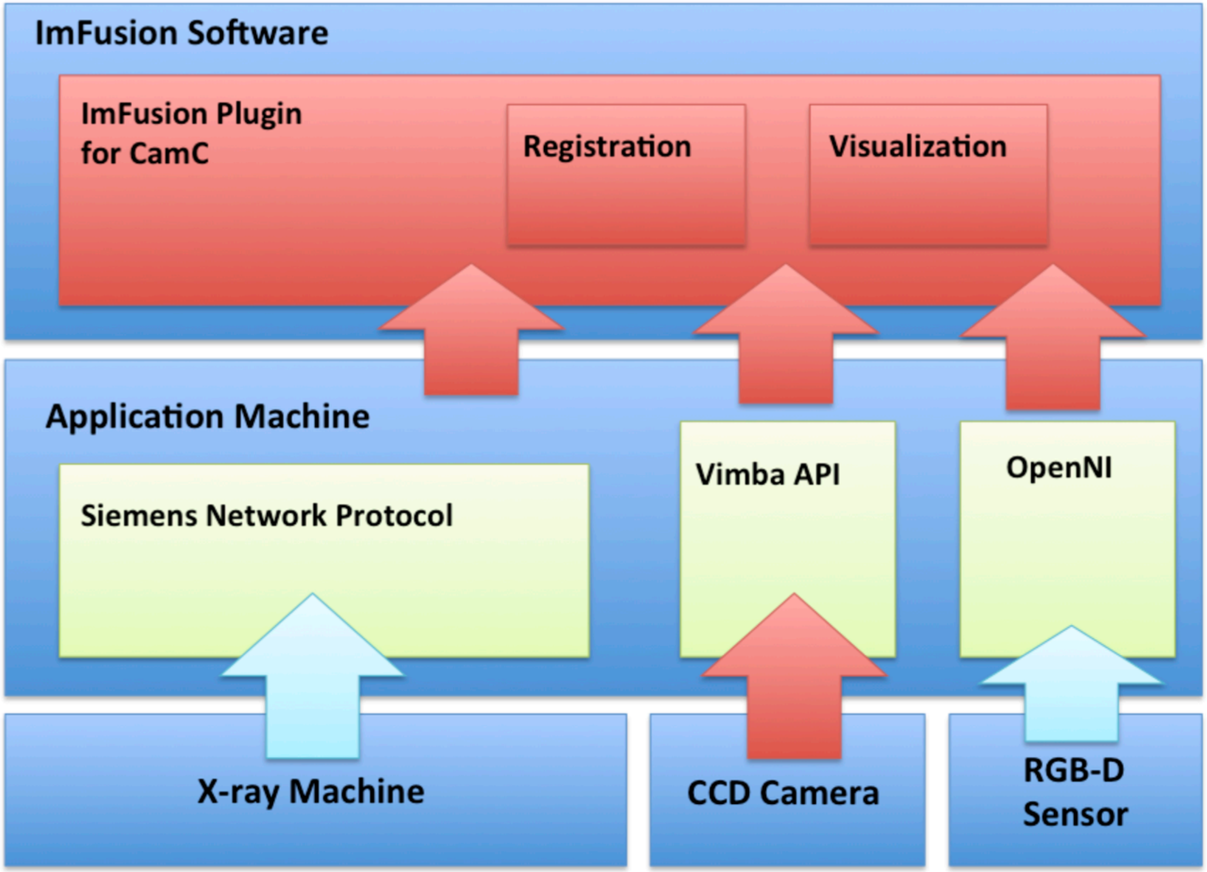
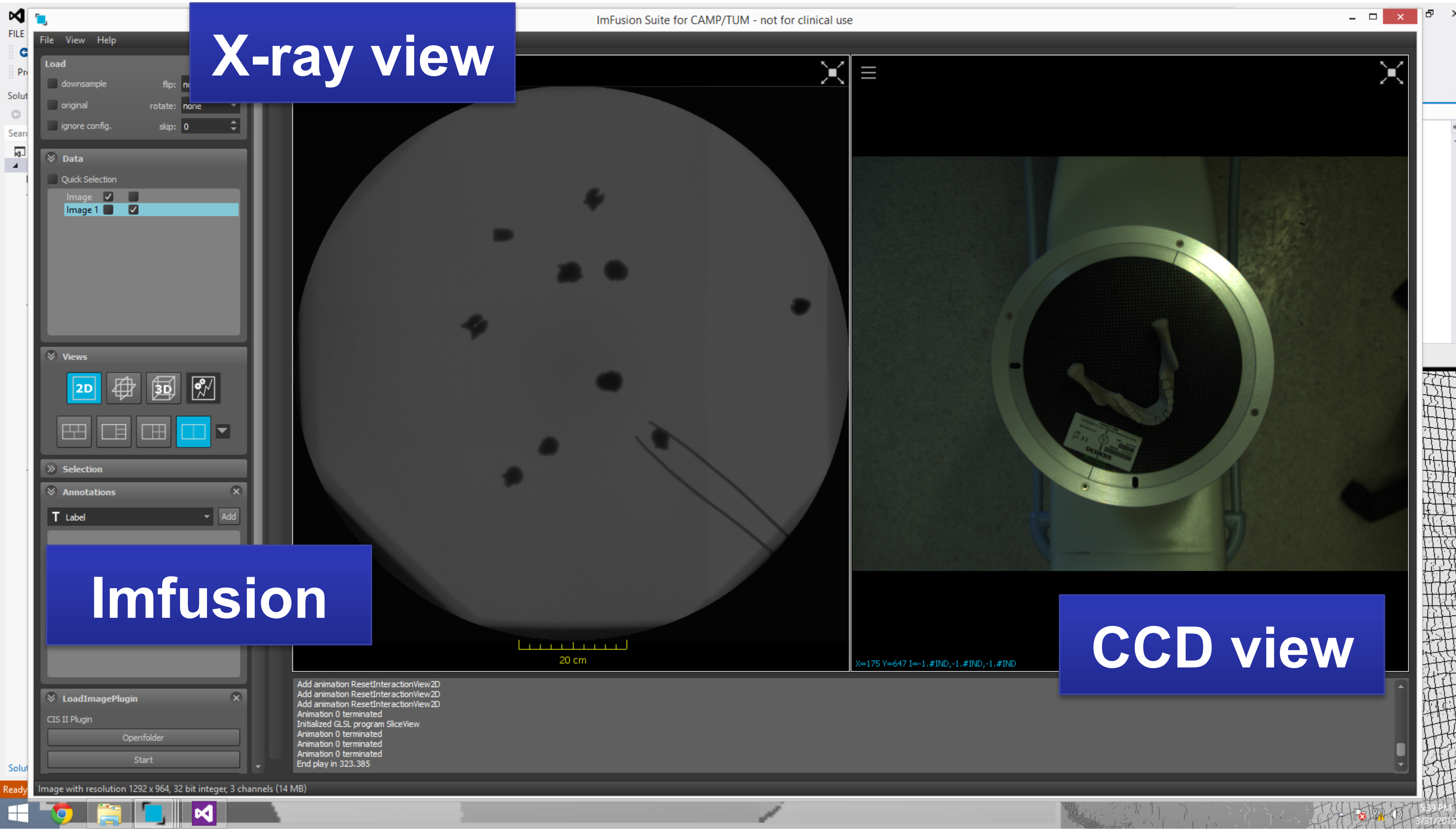


Figure 5. Software architecture block diagram


X-ray view
Imfusion
CCD view
ImFusion Suite for CAMP/TUM - not for clinical use
File View Help
Load
downsample
original
ignore config.
rotate: none
skip: 0
Data
Quick Selection
Image
Image 1
Views
Selection
Annotations
Label
Add
LoadImagePlugin
CIS II Plugin
Openfolder
Start
20 cm
X=175 Y=647 I=-1.#IND,-1.#IND,-1.#IND
Add animation ResetInteractionView2D
Add animation ResetInteractionView2D
Add animation ResetInteractionView2D
Animation 0 terminated
Initialized GLSL program SliceView
Animation 0 terminated
Animation 0 terminated
Animation 0 terminated
End play in 323.385
Image with resolution 1292 x 964, 32 bit integer, 3 channels (14 MB)

4. Registration

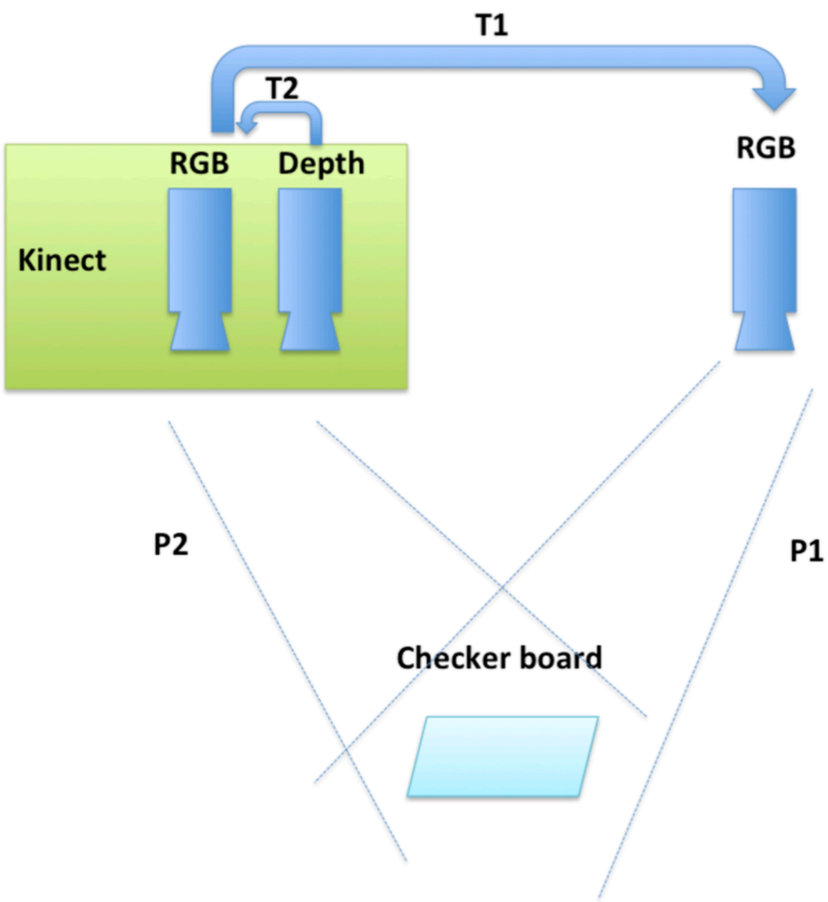


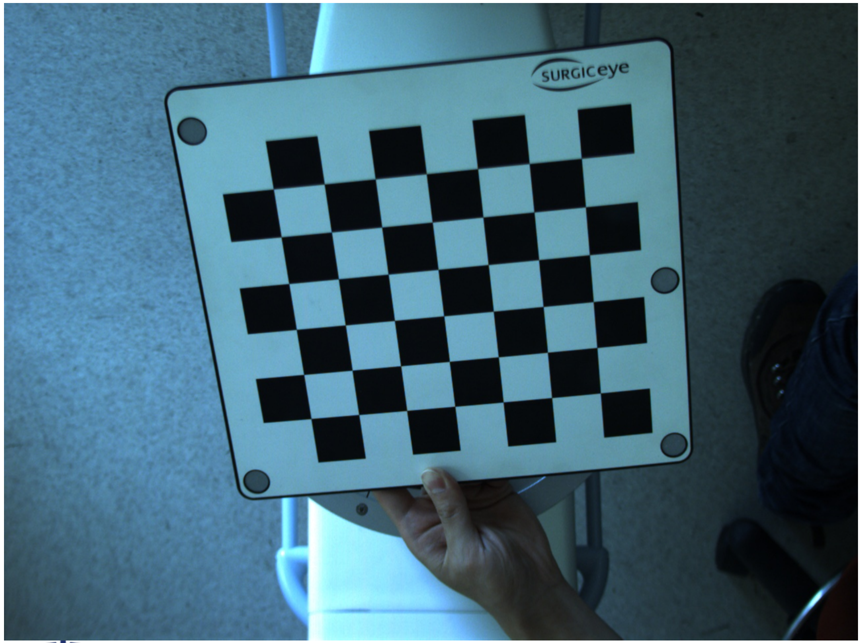
Figure 4. RGBD and RGB camera calibration

- Zhang, Zhengyou. "**A flexible new technique for camera calibration.**" Pattern Analysis and Machine Intelligence, IEEE Transactions on 22.11 (2000): 1330-1334.

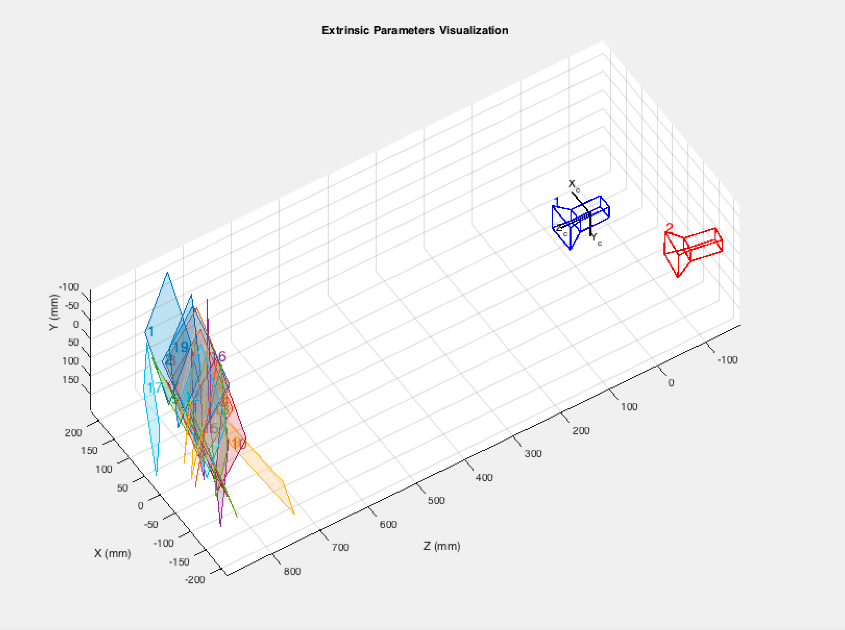
- Crop and up sample Kinect RGB and depth frame (bilinear sampling).

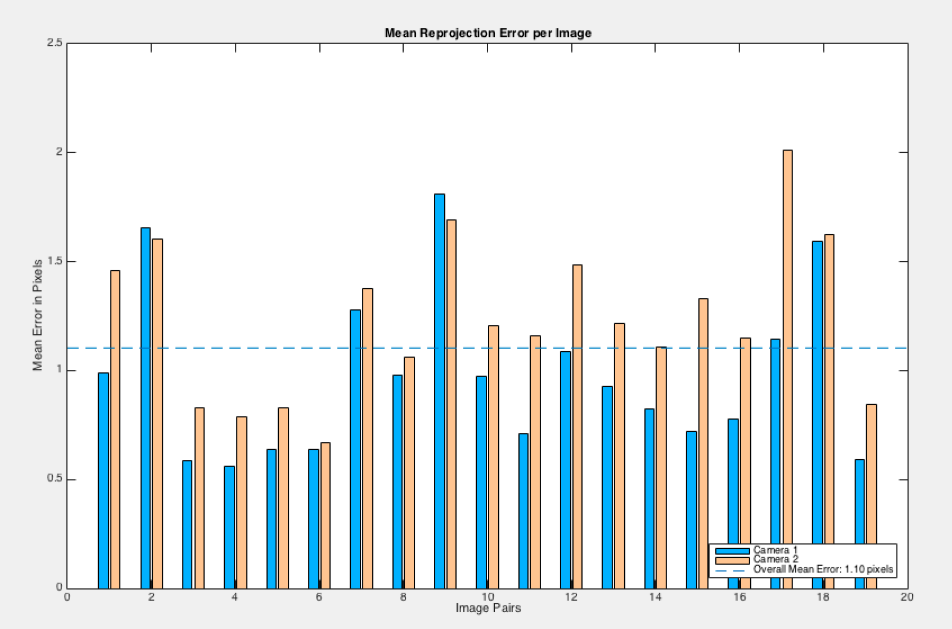
- Use the Matlab stereo camera calibrator to perform calibration.

- Extract intrinsic and extrinsic for future use.



Calibration result (resolution: 1292*964)





Math of back projecting

A pixel in the Kinect RGB image

$$p1 = (x\ y\ 1)^T$$

Transform into 3d with depth data

$$P1 = (X\ Y\ Z\ 1)^T$$

The extrinsic between two cameras

$$H = \begin{pmatrix} R_1^2 & T_1^2 \\ 0 & 1 \end{pmatrix}$$

Transform point into CCD camera frame

$$P2 = H * P1$$

Project 3d point into CCD image frame

$$p2 = \begin{pmatrix} \alpha & \gamma & ux & 0 \\ 0 & \beta & uy & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} * P2$$

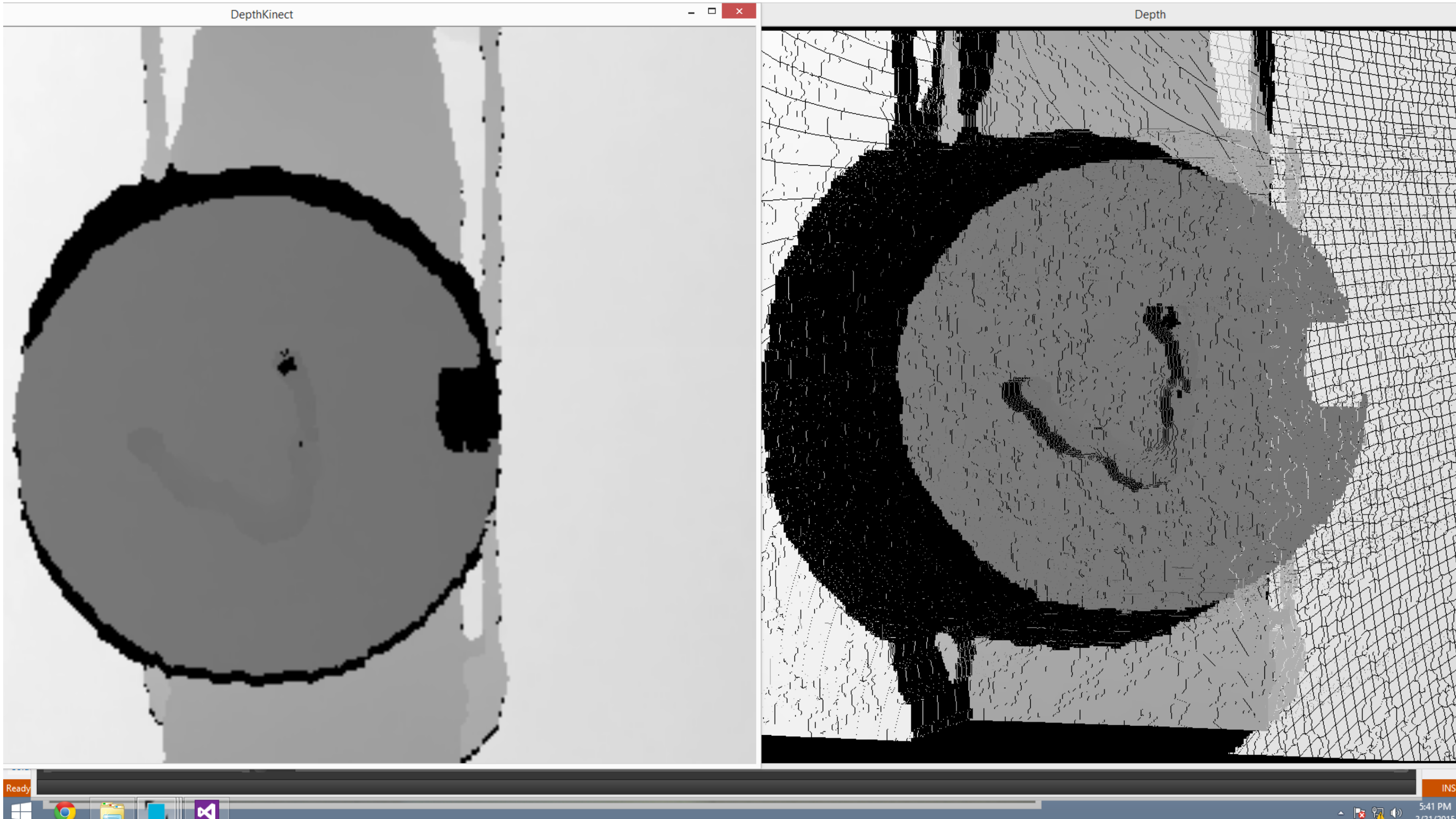
Scale p2 to make

$$p2 = (x_2\ y_2\ 1)^T$$

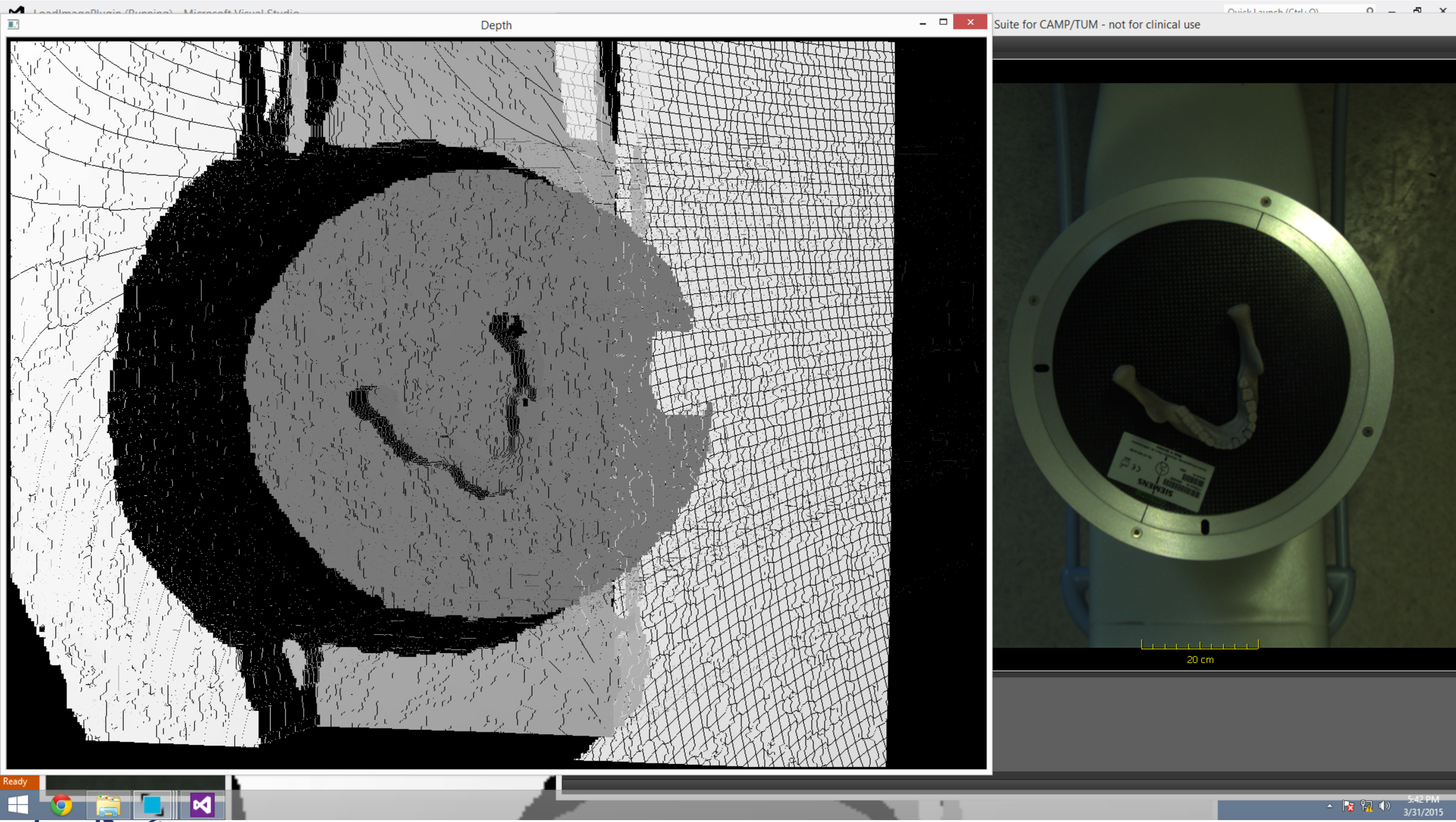
Pseudo code of back projecting (C++ with OpenCV)

```
while(running){

    cv::Mat KinectDepth, KinectRGB, CCDRGB;

    acquire(KinectDepth, KinectRGB, CCDRGB);
    crop_upsample(KinectDepth);
    crop_upsample(KinectRGB);

    std::vector<cv::Point3f> pointcloud;
    computepointcloud(pointcloud, KinectDepth);

    cv::Mat Rotation, Rotation_Vector, Translation, Intrinsic, DistortionCoefficient;

    cv::Rodrigues(Rotation, Rotation_Vector);

    std::vector<cv::Point2f> imagePoints;
    cv::projectPoints(pointcloud, Rotation_Vector, Translation, Intrinsic, DistortionCoefficient, imagePoints);

    cv::Mat CCDDepth;

    for(int i=0;i<imagePoints.size();i++){
        if((int)imagePoints[i].x >=0 &&(int)imagePoints[i].x <1292 && (int)imagePoints[i].y >=0 && (int)imagePoints[i].y <=964){
            CCDDepth.at<unsigned short>((int)imagePoints[i].y,(int)imagePoints[i].x) = (unsigned short)pointc[i].z;
        }
    }
}
```

Result



Result



5. Issues Encountered

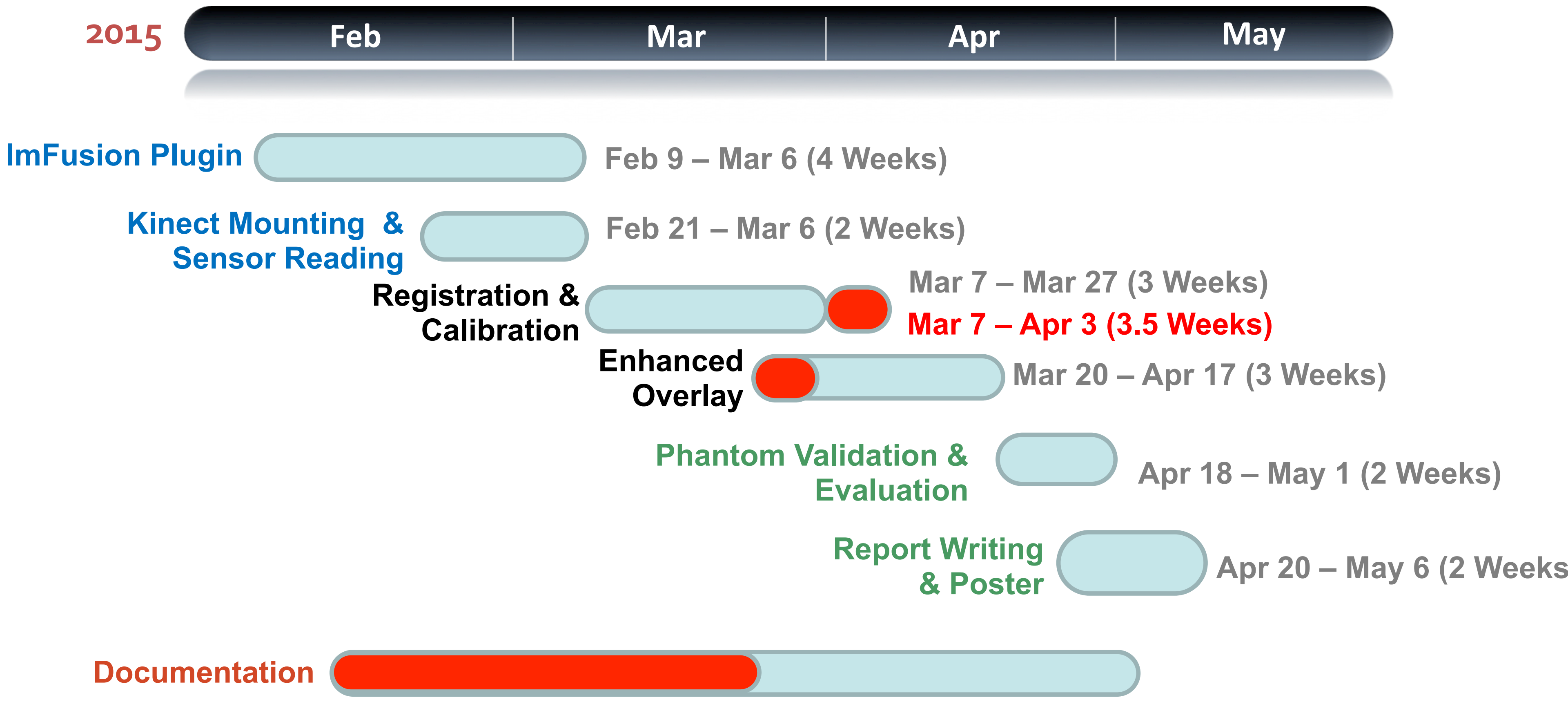
Most of the issues comes from hardware

- Low resolution of Kinect RGB and Depth Frame
  - Up sampling is a solution but affects accuracy
- Virtual Camera
  - Relationship original kinect camera
  - Point cloud computation
- Noise in the new depth map
  - Need analyze the source of noises
  - Need an algorithm to filter out noises
- Artifacts and occluding areas
  - Focus only on circle area
  - Increase kinect distance, or two kinect(future work)
- Speed
  - Parallelization
  - Texture mappingOpenGL
  - Better PC

6. Dependencies

- PC and remote control of C-arm application machine
  Expected resolve date: February 20 resolved
- Kinect sensor and its mounting supports
  Expected resolve date: March 6 resolved
- ImFusion source code for point cloud data
  Expected resolve date: February 27 resolved
- Registration and calibration tools
  Expected resolve date: March 4 resolved
- Animal tissue specimen and phantoms
  After developing the new system, I need to do validation with phantoms on the new system. I will get animal tissue specimens from the CAMP group.
  Expected resolve date: April 22

7. Timeline


2015
Feb
Mar
Apr
May
ImFusion Plugin
Feb 9 – Mar 6 (4 Weeks)
Kinect Mounting & Sensor Reading
Feb 21 – Mar 6 (2 Weeks)
Registration & Calibration
Mar 7 – Mar 27 (3 Weeks)
Mar 7 – Apr 3 (3.5 Weeks)
Enhanced Overlay
Mar 20 – Apr 17 (3 Weeks)
Phantom Validation & Evaluation
Apr 18 – May 1 (2 Weeks)
Report Writing & Poster
Apr 20 – May 6 (2 Weeks
Documentation

8. Milestones

• **February 27**: Finish developing ImFusion plugin for X-ray image and video acquisition. Completed
• **March 6**: Kinect mounted on C-arm and get point cloud data from ImFusion. Completed
• **March 27**: Kinect point cloud and video are registered; X-ray image and video are registered (Minimum deliverable achieved) Did not make it on that date
• **April 17**: An enhanced overlay developed (Expected deliverable achieved)
• **May 1**: Finish animal tissue specimen validation and evaluation (Maximum deliverable
achieved)
• **May 6**: Final poster presentation

Thanks for your attention!