

RGBD Camera Integration into Camera Augmented Mobile C-arm (CamC)

Project4 Final Report, EN.600.646 Spring 2015

Han Xiao

hxiao9@jhu.edu

Mentor : Dr. Nassir Navab, Bernhard Furest, Javad Fotouhi

May 7, 2015

1. Abstract

The project is based on the Camera Augmented Mobile C-arm (CamC); CamC is a system that fuses X-ray images and a real time video [1]. Without proper tracking, finding targets in X-ray view is difficult and requires multiple X-ray shots [2]. This will also increase the radiation exposure for both patients and surgeons. CamC provides an X-ray image overlay for surgical guidance with high accuracy, and can be used for many trauma and orthopedic surgeries. One limitation of CamC is the lack of depth information from the CCD camera view. In this project, the CamC is improved by integrating a depth camera. According to camera calibration and multi-view geometry, a depth map corresponding to the CCD camera is reconstructed, and consequently an improved X-ray overlay is rendered.

2. Statement of Purpose

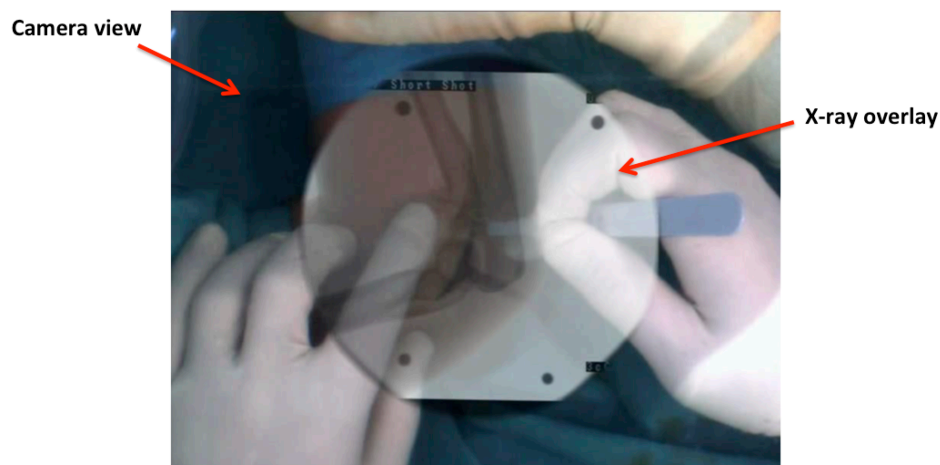


Figure 1. Overlay view of CamC. (Navab et al. IEEE TMI 2010)

The original view of CamC is illustrated in Fig.1. There is only one single CCD camera; therefore, X-ray overlay is always rendered on top of the optical video, which results in an unrealistic view. If the structure shown in the X-ray image is complicated in a procedure, such as a spine surgery, hands and tools will be hard to distinguish. This problem is shown in Fig. 2.

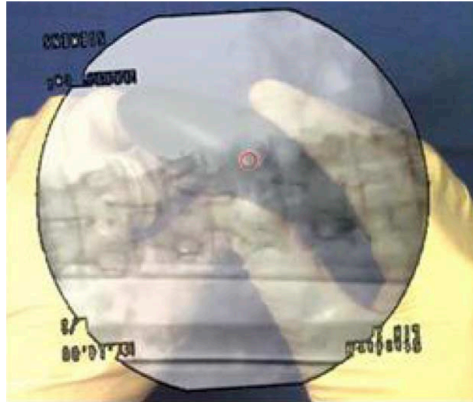


Figure 2. CamC in a simulated spine surgery. (Navab, Nassir, IEEE Transactions 2010)

A better visualization for the CamC system is needed to improve the usability. A method is needed to segment hands and tools, and tell spatial relationship between them and the patient. Traditional computer vision algorithms use intensity based segmentation, or optical flow to track hands and tools in the optical view. However, these algorithms have limitations running in the operation room because of the changing lighting condition, and they cannot provide spatial relationship. A depth sensor is used to reconstruct the depth data. Next, hands and tools are segmented using depth. An improved X-ray overlay is then created without having hands and tools blocked.

3. Technical Approach

3.1. Kinect Mounting

Kinect is used as the depth sensor, and the mechanical setup for the Kinect is illustrated in Fig. 3.

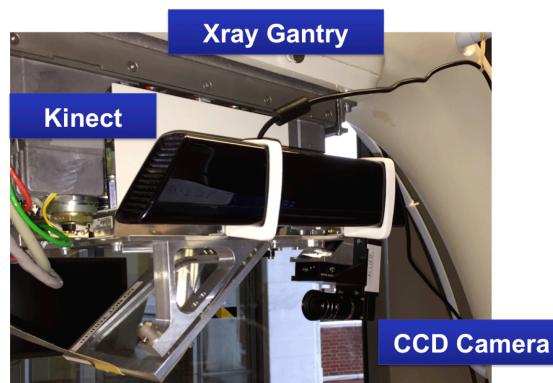


Figure 3. System setup.

A Kinect is mounted on the side of the X-ray gantry using 3D-printed mounts. The Kinect should have a distance from the image intensifier about 1 meter, which will reserve sufficient space

between objects being scanned and it. It will be positioned so that it can cover most space around the image intensifier.

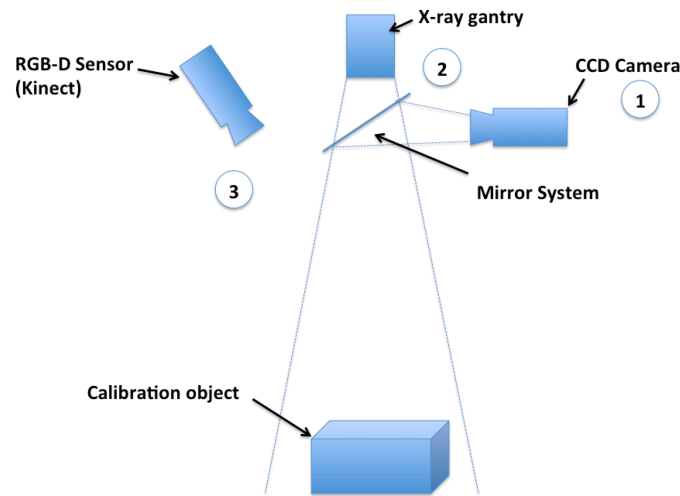


Figure 4. Overview of the camera system

3.2. Fusion of Optical and X-ray View

As illustrated in part 1,2 in Fig.4. There is a mirror system on CamC that align the axis of the x-ray gantry and the CCD camera. The system is already calibrated in the original CamC software. Using a calibration phantom with 5 pins, which are visible both in optical and X-ray view, the calibration is done with ICP [3] and Direct Linear Transform [4]. A homography is computed to register the optical and X-ray view. (Fig. 5)



Figure 5. X-ray and camera calibration

3.3. CCD camera and Kinect calibration

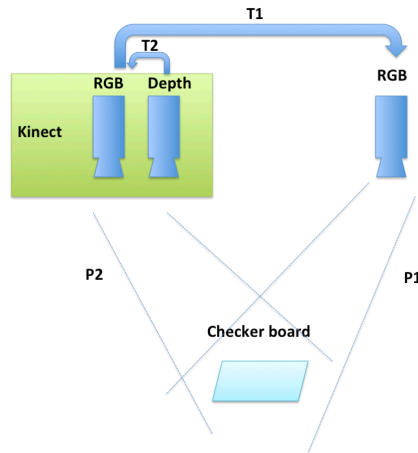


Figure 6. Kinect and CCD camera calibration

3.3.1. Distortion correction: A nonlinear radial distortion model is used to compute a lookup table for fast distortion correction for the CCD camera and the Kinect RGB camera.

$$X_{un} = D + X_{dist} \text{ with } X_{un} \in R$$

$$D = X_{dist}(k_1 r^2 + k_2 r^4 + \dots + k_i r^{2i})$$

X are points on the image. D is a polynomial function of the distortion coefficients k_i . The coefficients k_1 and k_2 are computed in calibration.

3.3.2. Kinect and CCD camera calibration: Checkerboard calibration technique is used to calibrate the Kinect and the CCD camera as illustrated in Fig. 7 in following steps. This method has already implemented in the Matlab's Stereo Cameras Calibration Toolbox using Zhang's flexible camera calibration technique [5].

- Scan a checkerboard in multiple positions with the Kinect and the CCD camera from two different views.
- Extract interest points on the checkerboards.
- Compute intrinsic parameters and transformation between two cameras (T1 in Fig. 6)
- The Kinect depth and RGB data is already co-registered in the OpenNI API (T2 in Fig. 6).

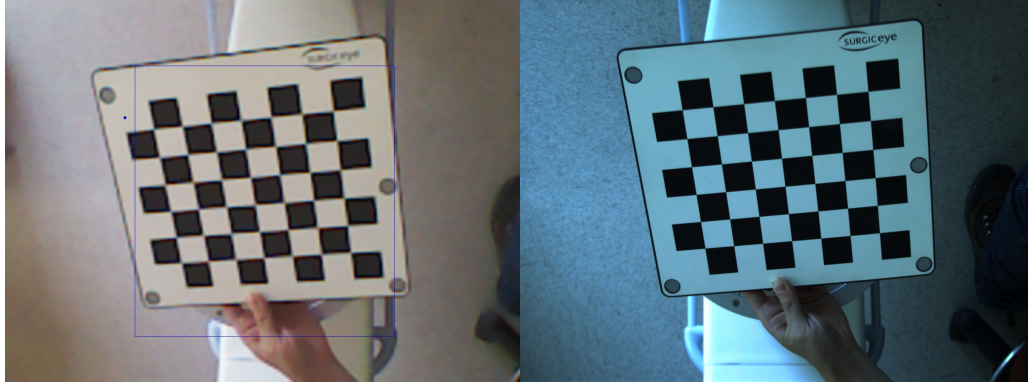


Figure 7. Checkerboard images for two cameras

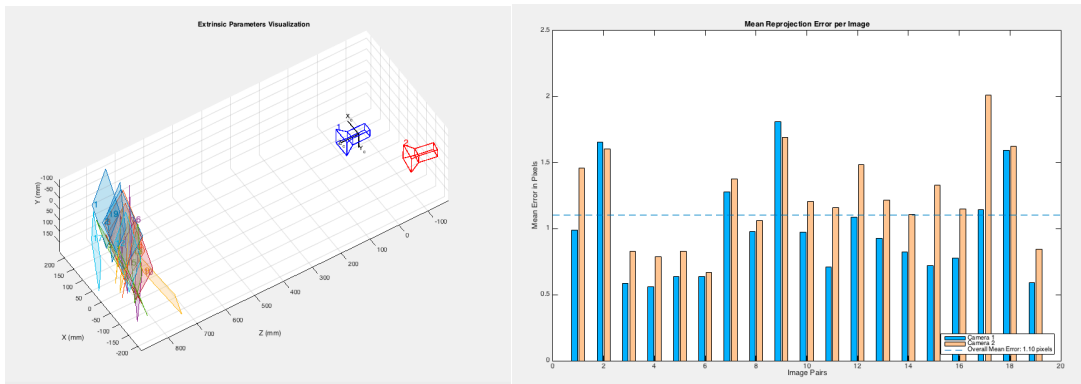


Figure 8. Calibration results

3.4. Depth map reconstruction for CCD camera

The depth data is registered with the RGB image in Kinect. A point cloud in the Kinect coordinate frame is then computed.

$$P_{kinect} = (X Y Z 1)^T$$

$$P_{CCD} = H * P_{kinect}; \quad H = \begin{pmatrix} R_{Kinect}^{CCD} & T_{Kinect}^{CCD} \\ 0 & 1 \end{pmatrix}$$

Knowing the extrinsic of the two cameras, the points cloud in Kinect coordinate frame is transformed into the CCD camera coordinate frame. H in the above equation is a homogeneous transformation with rotation R_{Kinect}^{CCD} and translation T_{Kinect}^{CCD} extracted from the camera extrinsic.

$$p_{CCD} = T_{3D}^{2D} * P_{CCD}; \quad T_{3D}^{2D} = \begin{pmatrix} \alpha & \gamma & ux & 0 \\ 0 & \beta & uy & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Projection Algorithm:

```

depthmap_ccd = new Mat;
compute  $P_{CCD}$  from  $P_{Kinect}$  point cloud;
for each point in  $P_{CCD}$  cloud
    compute  $p_{CCD}$ ;
    depthmap_ccd.at( $p_{CCD}$ ) = point.z;
end for

```

After projection, a depth map for the CCD camera view is reconstructed. However, there are problems with this depth map, as illustrated in Fig.9. The resolution of Kinect depth image is lower than the CCD camera image. A bilinear upsampling is performed on the Kinect depth map before doing point cloud transformation. However, the projection is not one to one mapping. After rounding projected pixel values, the projected depth map has pepper salt noises.

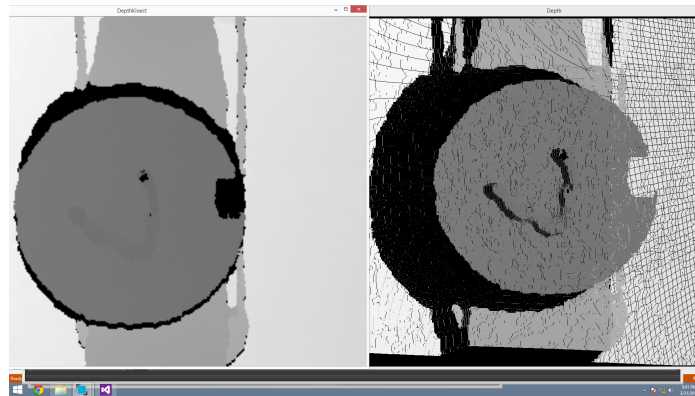


Figure 9. Original and reconstructed depth

The problem is improved by downsampling the Kinect depth map to do transformation and projection, then, upsampling the projected depth map for the CCD camera view. This is not the best way to solve the problem; a more accurate back projection method can be implemented in the future.

3.5. Enhanced X-ray overlay rendering

An enhanced X-ray overlay is rendered by the following steps: 1) Take an X-ray image and at the same time store the depth mesh of the phantom as a static reference. 2) Track the depth video stream and find out the depth changes. 3) Compare depth to the reference, create a mask and render overlay using the mask.

$$Mask = (DepthMap_{current} - DepthMap_{base}) > 0$$

4. Software architecture:

Software for CamC in the ImFusion [6] platform is developed in this project. The system diagram is shown in Fig. 10.

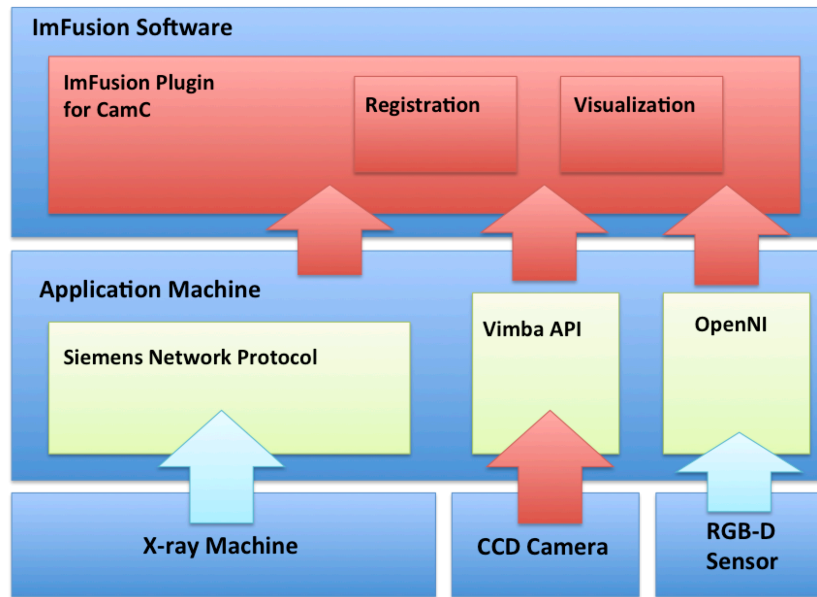


Figure 10. Software architecture block diagram

- **Application machine:** A fast PC runs software communicating with multiple devices.
- **Siemens network protocol:** Software developed by Siemens that dumps new X-ray images to a directory on application machine as raw files.
- **Vimba API:** API used to communicate with CCD Camera.
- **OpenNI:** Open source software for natural interaction devices. Used to communicate with the Kinect device.
- **ImFusion:** Powerful software solutions for medical image processing and computer vision.

The red blocks are parts developed in this project and other blocks are existing tools.

5. Results

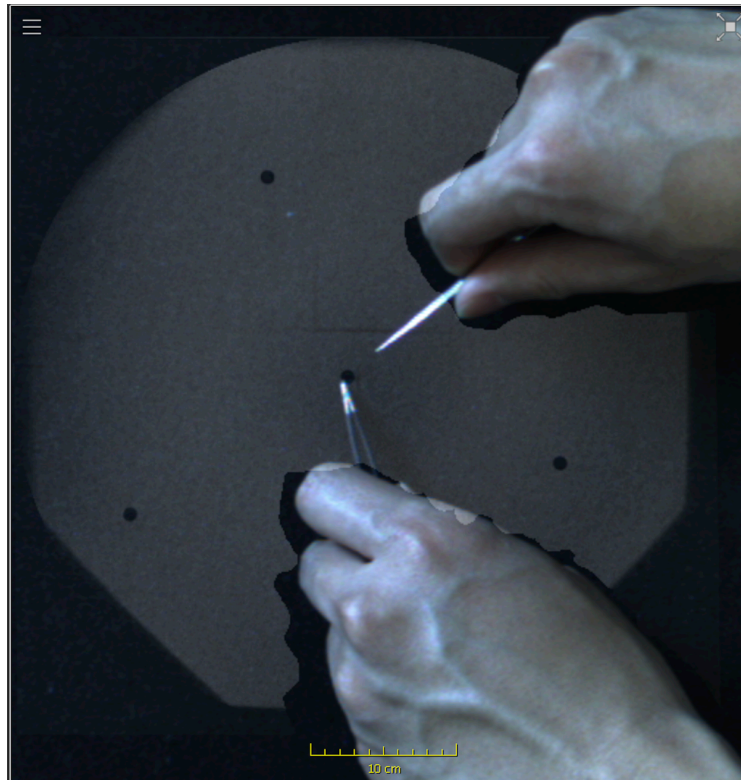


Figure 11. Improved X-ray overlay

Fig.11 shows the improved X-ray overlay of the calibration phantom. The hands are segmented and not blocked by the X-ray overlay. There are still some issues of this result. Because the depth image provided by Kinect sensor is flickering and has low resolution, the reconstructed depth image cannot provide perfect depth information, which results in artifacts around hand contours.

Software plugin is developed for ImFusion [6], which does:

- Multi-video capturing
- Processing and registration for X-ray image
- Depth map reconstruction
- X-ray overlay rendering

The data and functions in this ImFusion plugin are all reusable, and they can be easily used for further applications.

6. Deliverables and Timeline

6.1. Minimum (Achieved)

- ImFusion plugin for X-ray image acquisition, and CCD camera video acquisition.
- Kinect sensor mounting and point cloud acquisition.
- X-ray image – video calibration, and video – point cloud registration.

Goals: Generate a dynamic-link library file for ImFusion software, which creates a CIS II plugin. The plugin should have the following functions: 1) Monitor a local directory with X-ray image raw files, and extract new input images from the directory to ImFusion. 2) Read video from the CCD camera on CamC. 3) Read point cloud data from a Kinect sensor 4) Display a calibrated X-ray overlay on video, with live depth data for each pixel in the video.

6.2. Expected (Achieved)

- Enhanced X-ray overlay rendering

Goals: An enhanced X-ray overlay view should be displayed, which can adjust its transparency when there are surgeons' hands or surgical tools in the view.

6.3. Maximum (Uncompleted)

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

Goals: Invite a resident to perform simulated surgery on the new system with animal tissue specimens. Evaluate the procedure; compare it with existing procedures, and CamC procedure without depth information.

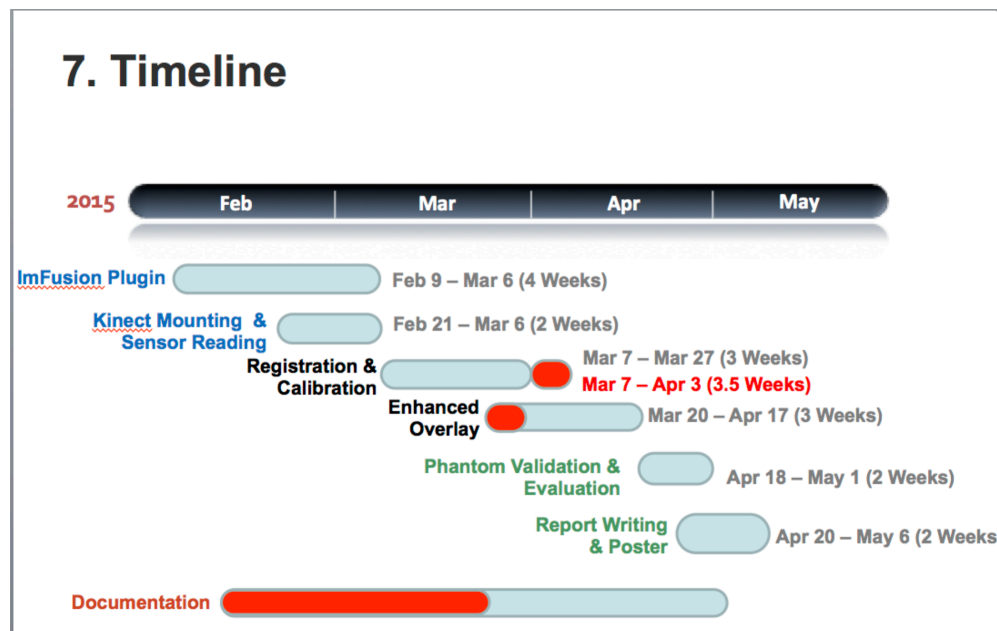


Figure 11. Timeline at mid term checkpoint

7. Timeline

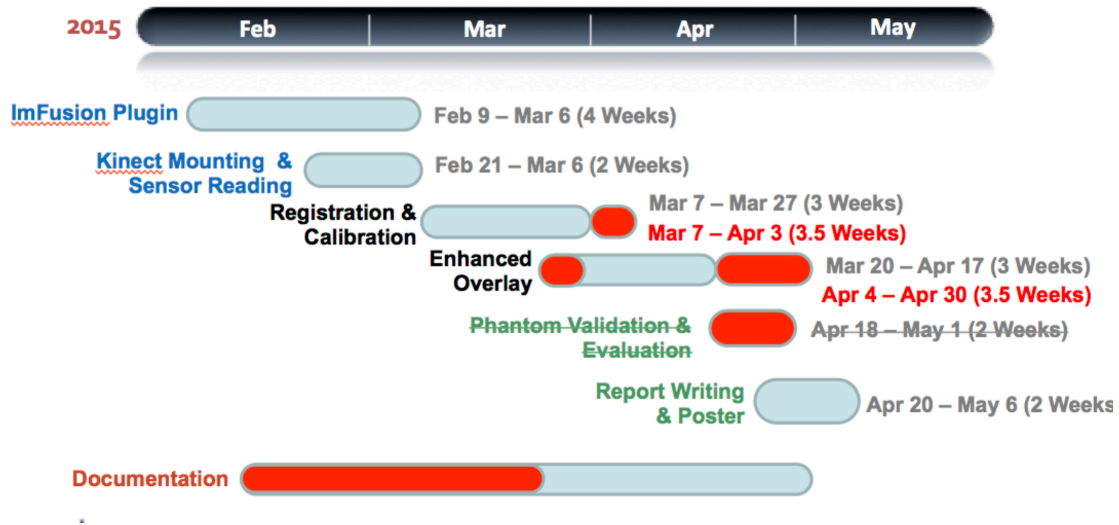


Figure 12. Final time line

Red block are tasks that are delayed

7. Future Work

- Usability study to evaluate enhanced visualization.**
 An evaluation process has not been set up for the new system due to time issue; evaluations for perception and user experience will be performed in the next step.
- Better depth interpolation and blending.**
 In this project, a simple bilinear sampling is used to interpolate depth map. In the future, a better interpolation technique is needed, and use back projecting technique for depth map projecting will generate better results.
- Parallel Implementation on GPU for speed-up.**
 The current software did computation mainly on CPU, some parts of the code came be implemented with GPU programming to accelerate computations.
- Further applications of depth data.**
 With the depth map, more applications can be developed, such as tracking, surface reconstruction during conebeam CT scan.

8. References

- [1] Navab, Nassir, S-M. Heining, and Joerg Traub. "Camera augmented mobile C-arm (CAMC): calibration, accuracy study, and clinical applications." *Medical Imaging, IEEE Transactions on* 29.7 (2010): 1412-1423.
- [2] Hofstetter, R., et al. "Fluoroscopy as an imaging means for computer-assisted surgical navigation." *Computer Aided Surgery* 4.2 (1999): 65-76.
- [3] Besl, Paul J., and Neil D. McKay. "Method for registration of 3-D shapes." *Robotics-DL tentative*. International Society for Optics and Photonics, 1992.
- [4] Richard Hartley and Andrew Zisserman (2003). *Multiple View Geometry in computer vision*. Cambridge University Press. ISBN 978-0-521-54051-3
- [5] Zhang, Zhengyou. "A flexible new technique for camera calibration." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 22.11 (2000): 1330-1334.
- [6] <http://www.imfusion.de/>

Support by and Acknowledgements

- Thank you to Bernhard Fuerst, Javad Fotouhi, Francisco Pinto, and Singchun Lee for providing help in system setup, software tutoring and algorithm development.
- Thank you to Dr. Nassir Navab for providing supports and the original CamC idea.