University of Western Ontario

Software Specifications: Camera Simulation Environment

Version 0.01

Author:
Kartik Thakore
kthakore@uwo.ca

Supervisors:
Dr. Hanif Ladak
Dr. Terry Peters

January 23, 2012

Contents

List of Figures

Glossary

Application Server An web service provider that responds to requests from the client. 6

HTML5 Is the fifth version of the Hypertext Markup Language that provides many new features such as canvas and web sockets. 6, 11

HTTP HyperText Transport Protocol defines the protocol of transporting documents to a web client. 6

Javascript A cross platform scripting language that runs on the client side. 6

OpenCV A software library that has a camera calibration component.. 6, 9

Pinhole Camera Model The pinhole camera model defines a camera as a mathematical relation between the coordinates of a 3D point, projected onto an image plane. This is also called the Perspective Model. It is a common camera model used in computer vision. 4

SIT System Integration Testing, test the overall system via, given inputs and expect outputs. 5

SRS Software/System Requirement Specifications. 11

UAT User Acceptance Testing, tests the system to ensure the user experience and expectations are met. 5, 7, 12

VR Virtual Reality: is a simulated environment that is implemented to provide a simulated experience of a real world process or activity. 4–6, 9

Web Browser A cross platform software package to view HTTP provided content from service providers. 6

WebGL A web technology being developed by the Khronos Group to bring accelerated graphics to the web browser, on a canvas. 6

Chapter 1

Introduction

A core component of Medical Imaging application is the use of computer vision. Camera calibration is a common task required for computer vision. Camera calibration can be a difficult task for researchers without prior graphics background. Cameras have been model in extensive literature that model, and thus can be simulated. A virtual environment would provide a activity based learning approach for camera calibration.

1.1   Overview

1. A virtual reality (VR) environment that simulates extrinsic and intrinsic parameters of a camera will provide a sufficient model to allow training on transformation matrices involved with camera calibration.

2. Camera calibration of a simple camera system, performed with prior training in a virtual reality environment, will yield insignificantly different capture camera parameters with real world camera system.

1.2   Background and Significance

Medical Imaging tasks such as image fusion, registration, feature tracking, etc. are built on the understanding of the mathematical concepts of vision systems. Computer vision is an increasing component of image-guided interventions. Moreover it is often difficult for researchers not from a computer vision background to learn quickly. Virtual environments that simulate computer vision systems would both hasten the learning curve and also act as a prototyping environment for new applications of video-based technology in image-guided interventions.

Mathematically, vision systems can be represented as matrices that perform operations on a spatial field. Cameras in medical imaging essentially transform three-dimensional content to two-dimensional planes (Pinhole Camera Model [2]). Thus, locations in 2D images correspond to the position and orientation of objects in the real world.

To be able to capture the position and orientation of objects, a transformation matrix is required from the 2D images. This transformation matrix is acquired by performing camera calibration. Calibration involves acquiring images of known objects with known positions and orientations in the real space. Next, feature points are selected on the 2D image. The selection of feature points are done either manually or using libraries such as OpenCV. The correspondence sets up the transformation between the real world coordinate system and the image coordinates.

Simulation of this process can begin by modeling the camera using the perspective projection model. The model can be represented in matrix notation as:

$$s * p = A * [R|t] * P \tag{1.1}$$

where A is defined as the intrinsic matrix, s represents the arbitrary scale factor, p are the 2D image coordinates, P are the corresponding 3D world coordinates, and finally $(R|t)$ are the extrinsic parameters.

The intrinsic matrix can be calculated from camera dependent parameters such as focal length and CCD pixel. Additionally, the extrinsic parameters are user defined such that rotation and translation are selected by the user. Finally, the P world coordinates are simply defined by the user and the scene that will rendered in the virtual reality environment.

A key challenge faced by graduate students in areas such as image-assisted surgery is in learning the calibration process and understanding the sensitivity of the calibration results on estimated parameters. Indeed, experience at the Robart's Research Institute indicates that students can take several months to learn about camera calibration. Simulation will allow for applications such as training and prototyping.

Once the prototype software system is developed, System Integration Testing (SIT) will be performed. Additionally, validation will be performed with User Acceptance Testing (UAT) of a selected group of volunteers. This will aid in narrowing down the causes for difficulty with understanding camera calibration.

1.3 Relevance

Camera calibration is a difficult task and a widely recurrent task at Robart's Research Institute in Medical Imaging. Additional camera calibration is a means to an end in most projects and not the focus. The essential tasks involved for performing a camera calibration are similar across projects with potential for standardization and collaboration. In Dr. Peters' laboratory, a considerable amount of effort is placed in camera calibration in several research projects. Each project involves isolated or loosely collaborated work on camera calibration. No software currently exists to help with training of camera collaboration.

Selecting a camera is crucial to a medical imaging application. A virtual environment may help to explore and compare various camera systems based on parameters provided by manufacturers. Allowing researchers to objectively try systems rather then purchasing each system and performing manual calibration. This will facilitate prototyping and will be a boon for researchers in gaining an intuitive understanding into the critical factors of camera calibration and the impact on their application.

1.4 Scope

The scope of this project is primarily to create a virtual environment that can reasonably simulate common cameras used in medical applications. The system will be able to create cameras given parameters that can be used to view a scene and run camera calibration tests on. Once calibrated, the system will allow the user to capture virtual images.

1. Develop a VR environment that can calculate transformation matrices based on given locations, poses, and camera parameters.

2. Implement a VR environment that simulates camera calibration within user specified scenes. Develop an online tool that can be accessed by a wide audience in the medical imaging field.

1.4.1 Pilot Scope

The scope of this pilot is to explore and define more deeply the problems with camera calibration, and a possible solution.

1.5 Technical Overview

1.5.1 User Interface

One of the initial aim of this project was to provide a easily distributed user interface. A web interface is a proven technology in providing distributed interfaces. The user interface provided by this system will be designed on a HTML5/WebGL project that will work over pre-specified Web Browser. WebGL sits on top of HTML5's canvas feature and will provide 3D rendering capabilities on enabled Web Browser. Most user demographics are familiar with using a Web Browser, which makes this a viable option.

WebGL will be written in the Javascript language that is a platform provided by most modern Web Browser.

An alternative option involves creating custom distribution and front end software (similar to a Web Browser). This is not feasible with the time frame for a Masters project.

1.5.2 Application Server

An Application Server will be built to provide the Javascript and HTML5 documents to the Web Browser. Additionally, the server will utilize middleware platforms that help to scale (ensure the Application Server responds to varying amounts of requests from multiple Web Browsers).

1.5.2.1 HTTP Server

The HTTP Server will be responsible for responding to requests sent by Web Browser clients. The key requirement required of the HTTP Server is to scale reasonably with an increase in demands. Most of the computational expense will be done on the Web Browser client, due to the way WebGL works. Therefore a scalable HTTP Server will ensure that the response for the WebGL and HTML5 documents are provided in a timely manner. For this reason the Nginx HTTP Server and a Starman/PSGI middleware server will be used.

A middleware server allows multiple Application Server to be initiated and used in parallel for times of load. This facilitates the reliability and scalability of the system.

1.5.2.2 Dispatch/Controller

The user will be using the system to do camera calibration that will require a interface between the frontend and OpenCV features. A controller will be used to define to protocol by which the frontend will communicated with the OpenCV Server.

1.5.2.3 OpenCV Server

This server will provide the Application Server with OpenCV features such as camera calibration and re-projection of calibration grids.

Chapter 2

User Specifications

This chapter defines the user of this system and their specifications. Defining the user ensure that the system can be designed with a UAT goal in mind.
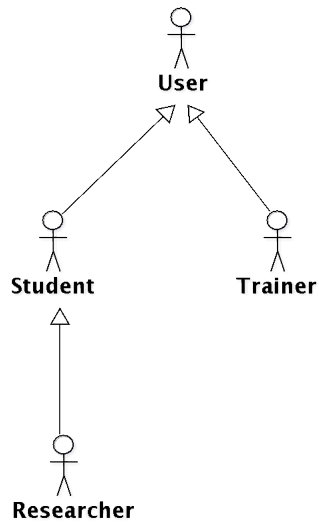

Figure 2.1: Users

2.1 Users

The system user is a generalization of several other users as depicted in the use case diagram 2.1.

2.1.1 Student

The student user is the primary user of the system. This system is essentially made to facilitate the understanding of the student user. Student users are defined as any user that needs to learn and work with camera transformation.

2.1.1.1 Researcher

The student user can also be extended as a researcher who needs a deeper understanding of camera calibration for a research project. These project involve camera calibration as a critical component in the implementation.

2.1.2 Trainer

Another user that may use the system is the trainer. Trainer users use the system as a guide for students and researcher. The direct requirements of the trainer user needs to be captured at a later date.

2.2 Use Cases

Use Cases define the process the system will take to meet the needs of the users. Each use case is written in a list format showing what the user will see and react with during the specific portion of the system.
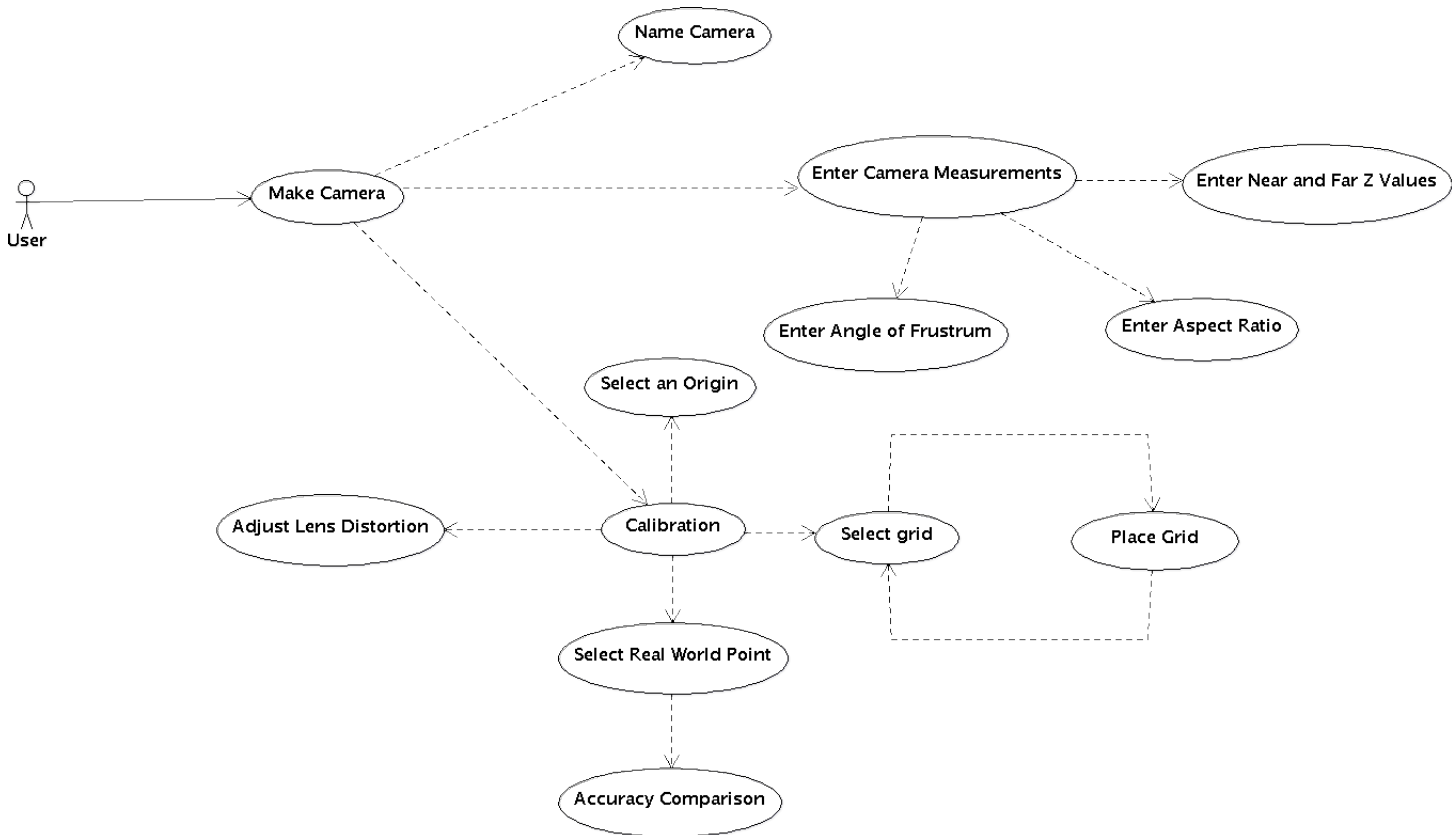
2.2.0.1 Camera Calibration

Figure 2.2: Generic User Cases

As described in 2.2, users can perform the camera calibration procedure as defined in Chapter 2 of [1]. All users of the system can perform a camera calibration procedure.

The first step involves creating a camera by with specified measurements. Users in future versions may select existing cameras that have been validated and stored on the system.
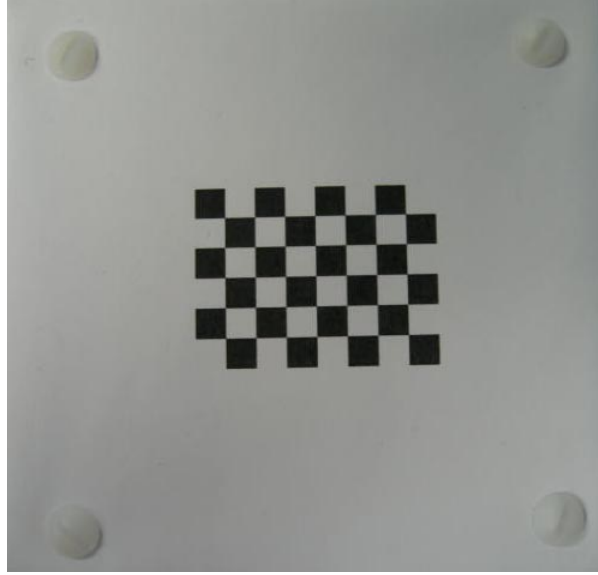

Figure 2.3: Calibration Grid –[1]

Using the camera the User can then select an origin point and place calibration grids 2.3 within the VR. Then by saving these images and analyzed for lens distortion (provided by OpenCV).

Next the user can then select a real world point, which is used to convey the final transform between the calibrated transforms and the real world transforms for their applications (depicted by $T_{R->S}$ in 2.4). Finally by adjusting the lens distortion parameters the user can compare the accuracy at certain points between two sets of distortion parameters.

2.2.0.2   Comparing Camera Systems

The Researcher user can perform calibration on two cameras and compare their accuracy models, to decided if they are suitable for their application. This use case is depicted in figure 2.5.

2.2.0.3   Training

Training involves two users. The Trainer can create a training session with a specified calibration procedure (camera, distortions, grid ... etc). Then the Trainer can send this session to a Student.

The Student will follow the Training Session and work towards a specified goal by the Trainer. The Student can also leave their feedback on completion that the Trainer can review, as described on 2.6

The Student completes the training session by finishing camera calibration and sending back the camera distortion parameters, extrinsic and intrinsic values to the trainer. Additionally the student can record the difficulty level of the steps of camera calibration. This feedback can be reviewed by the trainer.
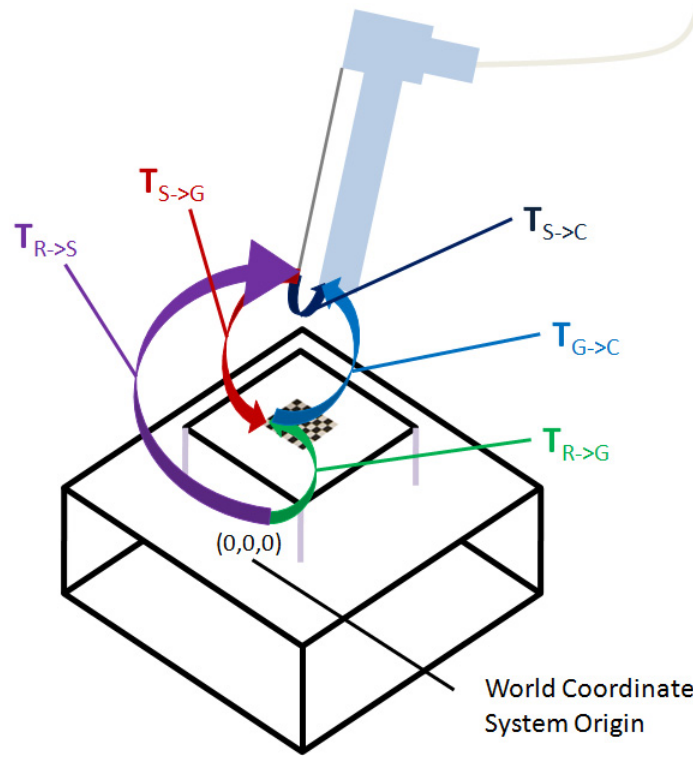
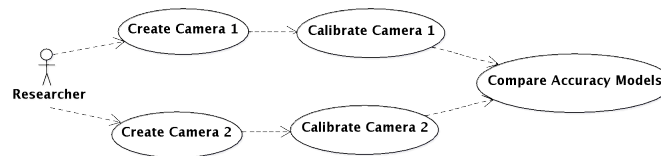Figure 2.4: Transformations involved during Calibration –[1]
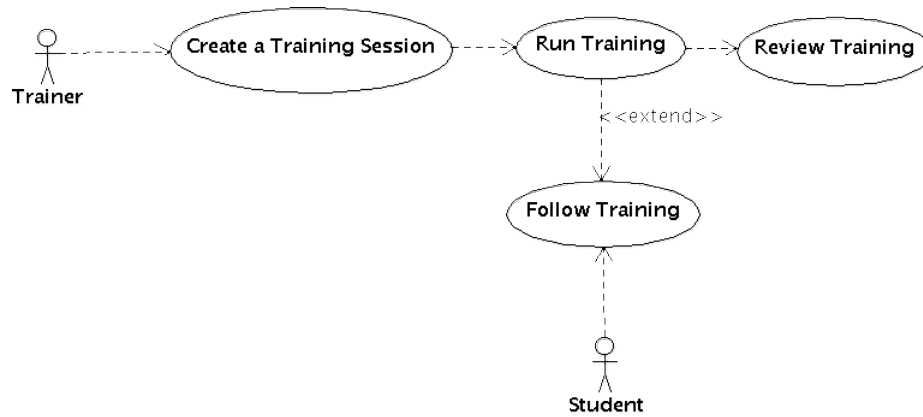

Figure 2.5: Researcher Use Cases

Figure 2.6: Generic Training User Cases

Chapter 3

System Requirement Specifications

The follow System Requirement Specifications (SRS) serve to outline a set of deliverables or goals for the prototyping period of analysis, design and implementation. These specifications are defined here to distill down the amount of uncertainty by picking standards were applicable. The initial round of testing on the prototype will help to clarify, extend and solidify the next iteration of specifications. This document is meant to modified and iterated.

3.1   User Interface

The user interface will be designed with standard HTML5 provided components for capturing user inputs, and providing feedback. These elements are well known and understood by a wide variety of users, reducing the steepness of the learning curve.

3.1.1   Interface Overview

There are no specific requirements for the user interface as of yet. However a general understanding of the screens has been captured as described in 3.1.
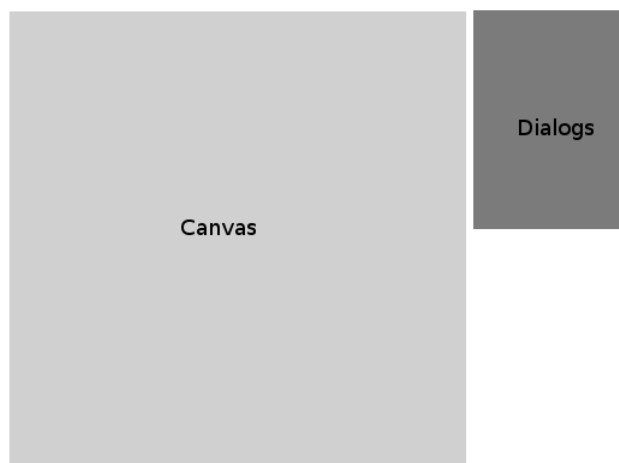

Figure 3.1: Interface Overview

The interface overview can be broken down into two sections. The canvas and the options menu. All 'work' related with either camera calibration will be visualized in the canvas section

11

of the interface. All inputs will be captured in the dialogs section of the interface. Finally any additional outputs will be show via the dialogs sections also.

3.2 Calibration

3.2.1 Interface

3.3 Training

The Trainer user can create a training session. A training session can consist of setting specific cameras, grids, and distortions. The trainer can then send a training plan to a student. The trainer will then review training.

3.3.1 Interface

3.3.2 Session

3.4 Constraints

3.4.1 Reliability

Reliability constraints are the same as expected of other web tools available. Industry standards of scalability are deployed into the system. Further testing will need to be performed to ensure expected reliability. This will be performed in UAT.

3.4.2 Scalability

Scalability constraints are the same as expected of other web tools available. Industry standards of scalability are deployed into the system.

3.4.3 Performance

Performance constraints are lax, as this is a prototype and performance profiles are not available.

3.4.4 Security

Currently there are no security constraints known.

Bibliography

[1] Carling L. Cheung, *Fusion of stereoscopic video and ultrasound for laparoscopic partial nephrectomy*, School of Graduate and Postdoctoral Studies, University of Western Ontario, London, Ont., 2010, 109 leaves ;;29 cm; Thesis (M.E.Sc.)–University of Western Ontario (Graduate Program in Biomedical Engineering), 2010.; Includes bibliographical references.

[2] David Forsyth and Jean Ponce, *Computer vision :a modern approach*, Prentice Hall, Upper Saddle River, N.J., 2003, David A. Forsyth, Jean Ponce.; :ill. ;;26 cm; "An Alan R. Apt book."; Includes bibliographical references (p. 643-671) and index.