Joshua Zapusek, Josue Perez, Aziz Boudy

CMPEN/EE 454

November 9, 2020

**Project 2: Camera Projection Project**

Part 1: Project Summary:

In this project we had the task to perform forward and inverse camera projection and perform all sorts of calculations along the way. The process of forward camera projection is the action of taking a 3D image and perform all sorts of transformations so that it would produce a 2D image. Inversely for inverse camera projection it consists of taking that 2D image and perform transformations, (in this case triangulation) to reconstruct a 3D image form the 2D image. The 3D points are represented as joints (i.e., shoulders, knees, etc.).

Through the duration of this assignment, we expected to need to interpret the mocap dataset for joint data, parse camera parameters, use those parameters to project 3D data into pixels for two image coordinate systems (pinhole model), to reconstruct the 3D locations of points (joints) using triangulation, to compute the Euclidean distances for final analysis, and to compute the algorithm efficiency.

In addition, we were tasked with computing the epipolar lines of the different views in the pinhole model.

We expect to glean further MATLAB experience and a deeper, practical understanding of Module 2 topics, including stereo vision, perspective, and motion capturing.

Part 2: Procedural Approach:

Attached to this document is P2\_FlowChart.drawio. This document contains the general control flow of out MATLAB procedures. Below is an outline of each function withing main.

Main script:

Parameters:

Outputs:

Project3DTo2D:

Parameters:

Outputs:

Reconstruct3DTo2D:

Parameters:

Outputs:

FindEpipolarLines:

Parameters:

Outputs:

MeasureError:

Parameters:

Outputs:

**Design Decisions:**

Parameter Choice and Utilization:

To calculate the 3D=>2D transformations for vue2 & vue4, we simply passed the former along with our original world points to projeect3DTo2D function. This function was obviously called twice, once for each respective camera’s perspective. Each vue held essential fields for calculating a 2D projection (Kmat & Pmat).

Section 2.2 Q&A:

Method of Demonstration:

Relevant Equations:

Part 3: Experimental Observations:

Some of the experimental observations

Part 4: Quantitative Results:

The performance of the project was mainly evaluated by comparing and calculating the sum of squares difference between the original 3D points given before any forward and inverse camera projections against the 3D reconstruction after triangulation for our inverse camera projection.

Below, in Figure 1, we have provided error tables for each joint pair

Below, in Figure 2, we have plotted the total error of all frames in a given run

Part 5: Qualitative Results:

The performance of the project was mainly evaluated by comparing and calculating the sum of

Part 6: Algorithm Efficiency:

An image (below) indicates our algorithm efficiency. This reference was calculated using the MATLAB profiler output.

Part 7: Epipolar Visualization:

Below you can see the result when calculating the epipolar lines and placing them for both images.

Part 8: Team Members Contributions:

Joshua Zapusek:

Josue Perez:

Aziz Boudy: