Assignment A1: Camera Models

Clinton Fernandes

u1016390

01/26/2016

A1

CS 6320

1. Introduction

The purpose of this assignment is to get the basic understanding of how a camera captures an image of an object in the world frame, i.e., the projection of a world object onto the camera frame. Different functions are to be developed in matlab for this exercise, first to make the world objects and then to transform the world points into the camera frame to get the image. Also, some questions which can be answered through this exercise are:

- How does camera projects the world points onto the camera frame?
- How sensitive is the projection to the camera parameters?

2. Method

CS5320_camera

Here, the camera function is defined. It takes the input form the different parameter values of the camera. One of the input to the function is also the scene i.e., the world points of an object (such as a cube). The camera function takes the scene and gives a projection onto the camera frame.

```
function im = CS5320_camera(scene,α,β,Q,x0,y0,R,t)
```

The following algorithm used to get the image points is:

$p = \frac{1}{Z} * \mu * P$

```
%this represents the perspective projection equation
 where p is the image point P is the scene point
```

```
μ = [α*R(1,:)-α*cot(Q)*R(2,:)+x0*R(3,:)  , α*t(1,1)-α*cot(Q)*t(2,1)+x0*t(3,1);
     (β/sin(Q))*R(2,:)+y0*R(3,:)          , (β/sin(Q))*t(2,1)+y0*t(3,1);
      R(3,:)                              , t(3,1)];
```

For the answer to 'how sensitive is the projection to camera parameters',

- First we need to fix the set of parameter values.
- Run the CS5320_camera function with the scene (example: cube).
- Store the values of the points obtained from CS5320_camera function as known points.
- Then, for each parameter:
    - add Gaussian to a parameter noise using randn() function in matlab to the parameter.
        - `noise = mean + std_deviation.*randn(numSamples, 1)`
        - `mean =0 and std_deviation=0.2 (assumption)`
    - Run the CS5320_camera function with the cube as the scene for 100 trials.
    - Store the values of the points obtained from CS5320_camera function for each trial.
    - Compare these points to the set of known point and calculate the error.
    - Determine the variance and mean of the error.
- Observe and make inference.

3. Verification

CS5320_gen_R()

CS5320_gen_R with arguments u = [0;0;1] and theta = 0 gives the identity matrix i.e., Rotation about z-axis by an angle of 0 radians.

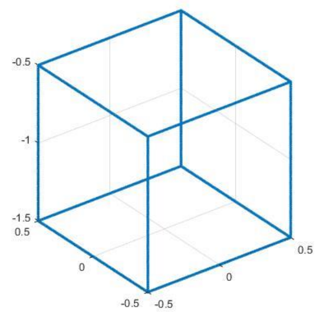
```
R =[t*ux*ux+C,t*ux*uy-S*uz , t*ux*uz+S*uy;
   t*ux*uy+S*uz,t*uy*uy+C , t*uy*uz-S*ux;
   t*ux*uz-S*uy,t*uy*uz+S*ux , t*uz*uz+C];

when u=[0;0;1] and theta=0
C=cos(theta=0)=1
C=sin(theta=0)=0
ux=0, uy=0, uz=1
t=cos(theta=0)=1

R =   1      0      0

      0      1      0

      0      0      1
```

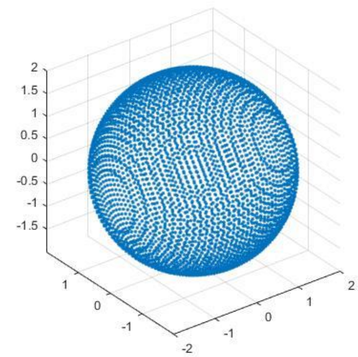
CS5320_gen_cube( )

Gives the points on the edges of the cube of the required size and centered wherever required in the Cartesian frame

CS5320_gen_sphere( )

Gives the points on the surface of a sphere of the required size and centered wherever required in the Cartesian frame. Distance between point along x axis is del_x and distance between two points on the surface is del_p.

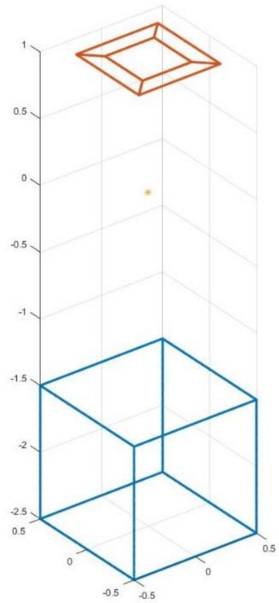


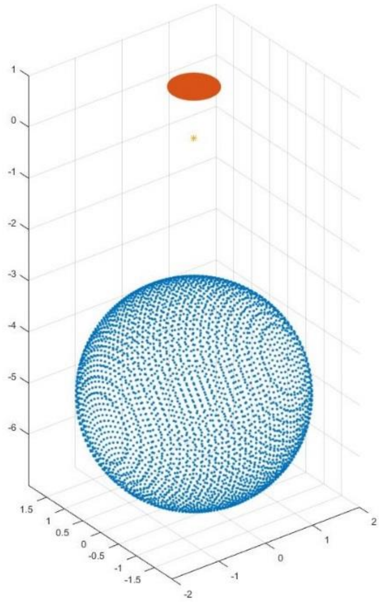


CS5320_camera( )

Gives the projection of the object's world points in the camera frame.

Example: in the view of the cube in z direction:





4. Data

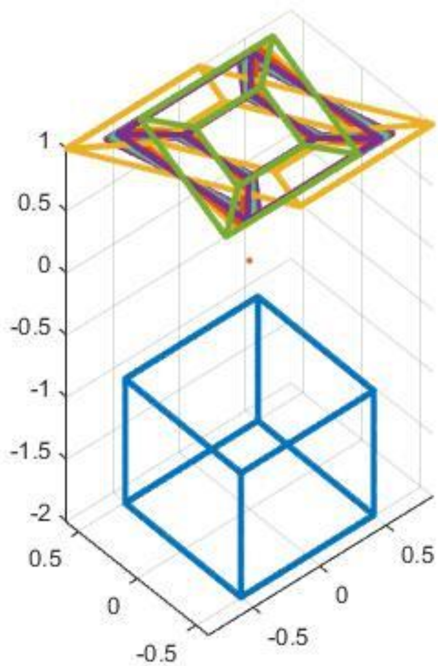
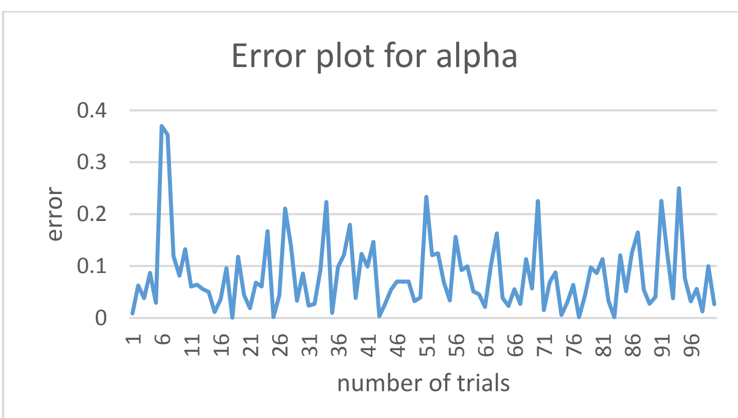
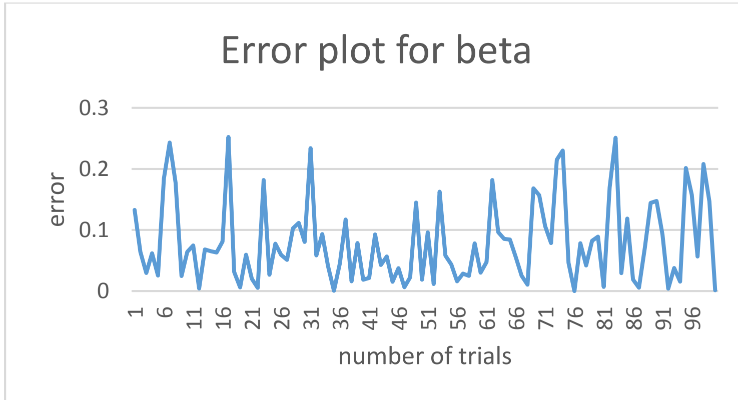
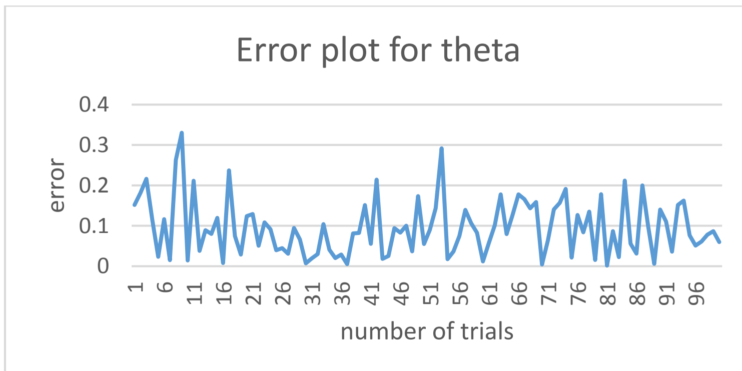


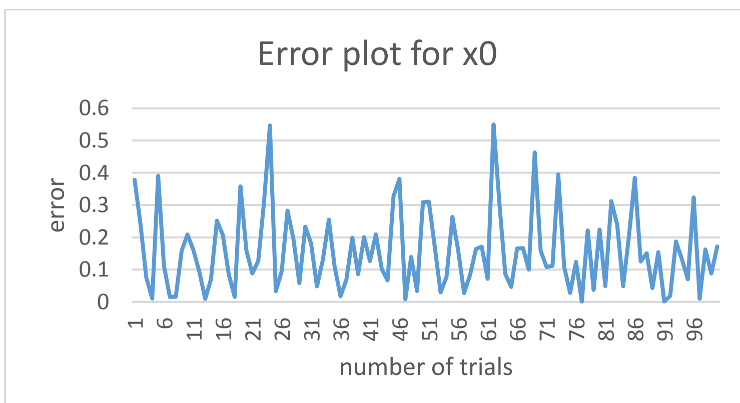
Figure: Images with Gaussian noise added to theta (10 trials)

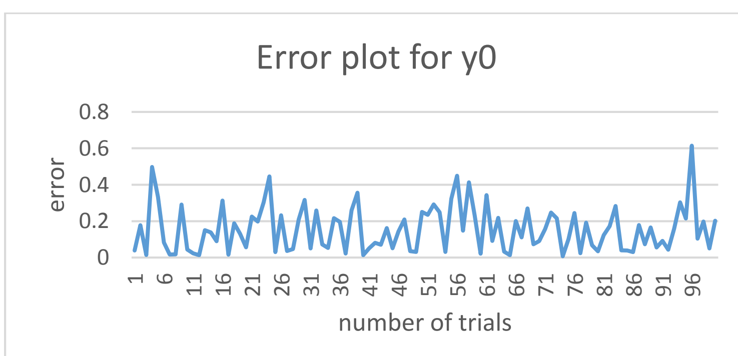
100 trials were carried out on the world points (cube with 1212 points). Following are the plots for error vs number of trials, when a Gaussian noise was added to each of the parameters. The mean of the noise was considered 0 and the variance was taken as 0.2.




Error plot for beta
error
0.3
0.2
0.1
0
1 6 11 16 21 26 31 36 41 46 51 56 61 66 71 76 81 86 91 96
number of trials


Error plot for theta
error
0.4
0.3
0.2
0.1
0
1 6 11 16 21 26 31 36 41 46 51 56 61 66 71 76 81 86 91 96
number of trials


Error plot for x0
error
0.6
0.5
0.4
0.3
0.2
0.1
0
1 6 11 16 21 26 31 36 41 46 51 56 61 66 71 76 81 86 91 96
number of trials


Error plot for y0
error
0.8
0.6
0.4
0.2
0
1 6 11 16 21 26 31 36 41 46 51 56 61 66 71 76 81 86 91 96
number of trials

5. Analysis

Parameter	Mean	Variance
Alpha	0.082194	0.004935
Beta	0.079072	0.004396
theta	0.095495	0.004819
x0	0.157886	0.014915
y0	0.1553	0.015646

6. Interpretation

It can be seen that the error due to the x0 and y0 parameters is much higher when compared to the error due to other parameters. This may occur as these parameters control the position of the image more so than other parameters.

The mean of error for the Alpha, Beta and theta parameters was found to be closely related. It is also observed that the variance in the image position closely resembles the variance of the Gaussian noise (0.04) when it is added to alpha, beta and theta parameters.

7. Critique

The experiment could be improved by taking a greater range in the Z coordinate.

It would also be good to see the error in the image with different mean and variance for the Gaussian noise

8. Log

Programming/Debugging: 14 hours

Report: 3 hours