Illumination and Projection Model
Project 01 - Report

Usama Munir Sheikh
ECSE 6650: Computer Vision with Dr. Qiang Ji, RPI
Troy, New York
sheiku@rpi.edu

***Abstract*— I have applied the full, weak and orthographic projection equations and the fundamental radiometric equation for illumination to produce 2D images of a 3D face under different camera parameters and illumination settings.**

Keywords—Affine Transformation; Focal Length; Full Projection Model; Orthographic Projection Model;Radiometric Equation; Weak Projection Model;

I. Introduction

Cameras are the most important sensor in field of computer vision and therefore, it is important to understand how a 3D scene in the world relates to a 2D image produced by the camera.

In this project I investigate projection models that relate the spatial coordinates of 3D points to 2D image points. The three projection models investigated here are the Full, Weak and Orthographic Perspective projection models. These models are based on the Pinhole Camera Model. The project requires us to apply the Full Perspective projection model.

A. Affine Transformation

The first part of the Full Perspective projection model is an Affine Transformation in homogenous form, that transforms the (X, Y, Z) coordinates of a 3D point in the object frame to the camera frame coordinates: (X_c, Y_c, Z_c). It involves a 3D rotation and translation.

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = R \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + T;$$

$$\Rightarrow \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = [R \quad T] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix};$$

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}; T = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

The rotation here is given by a 3×3 matrix, R and the translation is given by a 3×1 column vector, T. The rotation matrix can also be written as a 3×1 column vector of rows with each row being a 1×3 row vector themselves.

B. Perspective Projection

Then, we have the perspective projection part of the model that relates the 3D coordinates in the camera frame to the points on the image array, (u, v).

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}$$

where λ, is a scaling factor that depends on the type of projection model chosen. For the Full Perspective projection model, $\lambda = Z_c$ where Z_c is the z coordinate of the 3D point after the affine transformation, $[R \quad T]$.

For Weak Perspective *projection*, we use $\overline{Z_c}$, instead of Z_c, which is the average distance of the object to the camera along the optical(Z) axis (in the camera frame.)

For Orthographic projection, which is a unique case of Weak Perspective projection, this part changes. This is because, for the orthographic projection the ratio: $\frac{f}{Z_c} = 1$.

C. Sampling/Spatial Quantization

These points, (u, v), are then sampled and given the correct offset, (c_0, r_0)(also known as the principal point), in the final part to generate pixel locations: (c, r).

$$\begin{bmatrix} c \\ r \\ 1 \end{bmatrix} = \begin{bmatrix} S_x & 0 & c_0 \\ 0 & S_y & r_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

Combining all three parts gives us the final projection equation.

D. Projection Matrix

The final projection equation for Full perspective projection is given below:

$$\lambda \begin{bmatrix} c \\ r \\ 1 \end{bmatrix} = \begin{bmatrix} S_x f & 0 & c_0 \\ 0 & S_y f & r_0 \\ 0 & 0 & 1 \end{bmatrix} [R \quad T] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\Rightarrow \quad \lambda \begin{bmatrix} c \\ r \\ 1 \end{bmatrix} = WM \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$$\Rightarrow \quad \lambda \begin{bmatrix} c \\ r \\ 1 \end{bmatrix} = P \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

where, (c, r), are the pixel coordinates on the image plane, λ is a scaling factor and

$$W = \begin{bmatrix} S_x f & 0 & c_0 \\ 0 & S_y f & r_0 \\ 0 & 0 & 1 \end{bmatrix}; \mathrm{M} = [R \quad T]$$

$$P = WM$$

W is known as the intrinsic matrix and M is known as the extrinsic matrix because each embodies the intrinsic and extrinsic or external aspects of the projection model respectively. Multiplying the two together gives us P which is known as the projection matrix. The projection matrix incorporates all the three parts from before in itself. Multiplying a 3D point with P can directly give us the image pixel locations. This comes in handy during the implementation of the projection equation.

The projection matrix for Weak and Orthographic projection models is derived the same way except for the differences mentioned before when describing the role of λ.

The projection matrices for Full, Weak and Orthographic projection models are given below.

$$P_{full} = \begin{bmatrix} S_x f r_1 + c_0 r_3 & S_x f t_x + c_0 t_z \\ S_y f r_2 + r_0 r_3 & S_y f t_y + r_0 t_z \\ r_3 & t_z \end{bmatrix}$$

$$P_{weak} = \begin{bmatrix} S_x f r_1 & S_x f t_x + c_0 \overline{Z_c} \\ S_y f r_2 & S_y f t_y + r_0 \overline{Z_c} \\ 0^{1\times3} & \overline{Z_c} \end{bmatrix}$$

$$P_{orth} = \begin{bmatrix} S_x r_1 & S_x f t_x + c_0 1 \\ S_y r_2 & S_y f t_y + r_0 1 \\ 0^{1\times3} & 1 \end{bmatrix}$$

E. Illumination Model

Also important in computer vision, is the study of illumination. Illumination models are necessary to calculate the appropriate intensity values of individual image pixels based on light reflected by the 3D object. In this project we use the Fundamental Radiometric Equation to find these intensity values.

$$I = \frac{\beta \rho \pi}{4} \left(\frac{d}{f}\right)^2 cos^4(\alpha) L \cdot N$$

The above equation is the fundamental radiometric equation where I is the intensity of the pixel, L is the vector for the lighting direction and N is the normal vector for the 3D point.

F. Project Goals

The goal of this project was to apply the Full Projection Model and the Fundamental Radiometric Equation to obtain 2D images of the 3D face, shown in figure 1, under different parameters and discussing the results, specifically, analyzing the effects of each parameter on the resulting 2D image.

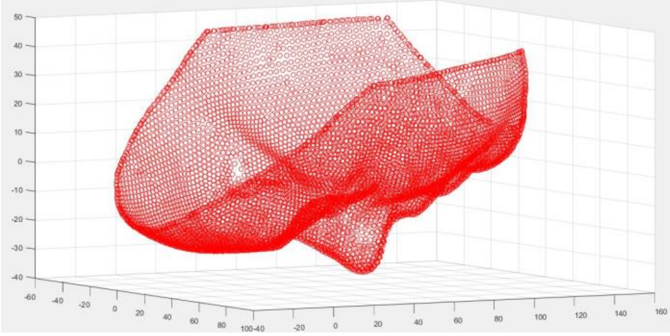


Figure 1. 3D Object (Face)

There were three sets of parameters that were varied between two different values for each, to get eight sets of images. The parameters that were varied include (1) the relative rotation between the object and the camera, (2) the focal length of the camera and (3) the lighting direction.

The effects of changing each of the parameters can be clearly seen in the results and clear comparisons can be made, enabling us to deduce how a particular set of parameters would change the 2D projection.

As an extra credit problem, I also obtained results for the Weak and Orthographic Projection Models.

II. Related Theories

There are many other illumination models out there that can be used other than the fundamental radiometric equation we used. These models are widely used for illumination in the field of computer graphics. [2]

The three most common models used are the 1) Ambient Illumination Model, 2) Diffuse Reflection or Lambertian Model and 3) Specular Reflection or Phong Model [2]. The radiometric equation that we use is based on the Lambertian Model.

The Ambient Illumination model is the default model used in OpenGL. It assumes there is some non-directional background light in the environment and it is the simplest model for illumination. The reflected lights intensity, I_{amb} at any point on the surface is given by

$$I_{amb} = K_a I_a$$

where, I_a is the ambient light intensity and K_a is the surface ambient reflectivity.[2]

The Diffuse Reflection Model works best on matte or Lambertian surfaces and it uses Lamberts cosine law for reflection. The surface appears equally bright from all viewing direction in the Lambert model. The reflected light intensity, I_{diff} at any point on the surfaces is given by

$$I_{diff} = K_d I_p \cos(\theta) = K_d I_p (N \cdot L)$$

where, I_p is the point light intensity, K_d is the surface diffuse reflectivity, N is the surface normal and L is the light direction. [2]

Lastly, there is the Specular Reflection Model that is used for shiny and glossy surfaces like metal, plastic etc. The reflected light's intensity, I_{spec}at any point on the surface is given by

$$I_{spec} = K_s I_p cos^n(\varphi) = K_d I_p (R \cdot V)^n$$

where, I_p is the point light intensity, $K_s$is the surface specular reflectivity, R is the reflected light direction for an ideal specular surface, V is the reflected light direction of the non-ideal specular surface and n is the specular reflection parameter that determines deviation from the ideal specular surface. [2]

We can also get an illumination equation by combining all three models linearly by assigning weights (according to surface type) to intensity values from each of the models and summing them together. [2]

III. Problem Statement and Methodology

To start the problem off, I was given the xyz coordinates of the 3D face as three arrays: X, Y and Z. Also, I was given the xyz components of the normal vector corresponding to each of these points as three arrays: $N_x, N_y \ and \ N_z$. The data was provided to me in the form of a *MATLAB* file and therefore the work was coded in *MATLAB.*

Also, I was provided with the Rotation and Translation Matrices for the Affine Transformation shown earlier. Two different rotation matrices, R_1 and R_2 were provided to compare their results.

$$R_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \ \& \ R_2 = \begin{bmatrix} 0.9848 & 0 & 0.17360 \\ 0 & 1 & 0 \\ -0.1736 & 0 & 0.9848 \end{bmatrix}$$

$$T = \begin{bmatrix} -14 \\ -71 \\ 1000 \end{bmatrix}$$

Also provided were two values of focal length and two lighting directions to see the effect of each parameter on the resulting 2D image.

$$f_1 = 40mm \ \& \ f_2 = 40mm$$

$$L_1 = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} \& \ L_2 = \begin{bmatrix} 0.5774 \\ -0.5774 \\ -0.5774 \end{bmatrix}$$

Other parameters provided were the spatial sampling frequencies, the principal point and parameters for the radiometric equation.

$$S_x = S_y = 8 \frac{pixels}{mm}$$

$$c_0 = 50 \ \& \ r_0 = 50$$

$$\alpha = 0; \ \beta = 1; \ \rho = 1; d = 33$$

Note here that, $\alpha = 0$, because for the pinhole camera model the term, $cos^4(\alpha)$, in the radiometric equation can be ignored because α is really small.

Using the given set of parameters, I implemented the Full Projection Equation (using the projection matrix, P) and the fundamental radiometric equation in *MATLAB.*

For each xyz coordinate of the 3D point given, I would compute the corresponding location, (c, r) using the projection equation (discussed earlier) and the corresponding intensity value, I at that pixel using the radiometric equation that was presented earlier.

The code for the whole project is given in the appendix, at the end of the report. There is only a single m-file that that user has to run in *MATLAB.* It first shows the scatter plot of the 3D face, as shown in figure 1 and then the program prompts the user for inputs: to select the type of projection model and to select between the two options for each of the parameters we are varying: the rotation matrix (R_1 or R_2), the focal length ($f_1 \ or \ f_2$)and the lighting direction($L_1 or \ L_2$).

One of the things to note here is that before the perspective equation is applied, the code multiplies the lighting direction vector, L and later it multiplies the normal vector, N with the rotation matrix to change their coordinates into the camera frame from the object frame. For N, this is done inside the loop because it has to be done for the normal vector of every 3D point before a dot product can be taken within the radiometric equation.

Also note that the code sets any negative intensity value to zero which gives some black holes inside the image. This is rectified by assigning an average of the neighboring pixel intensities to the black pixel.

IV. Experiments & Results

The program was run with all eight scenarios to generate $100 \ x \ 100$ images for each. The results for Full Perspective Projection Model are shown in figures 2 through figure 5.

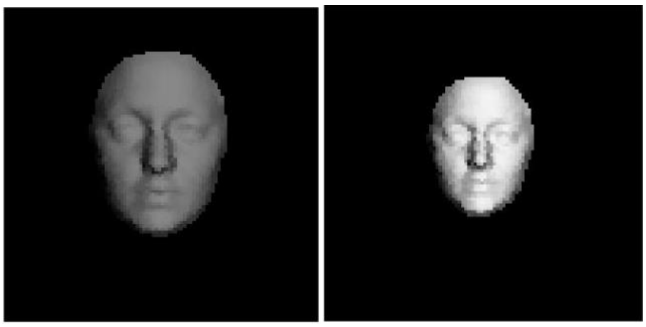


Figure 2. (Left) R1, L1, f = 40mm
(Right) R1, L1, f = 33mm

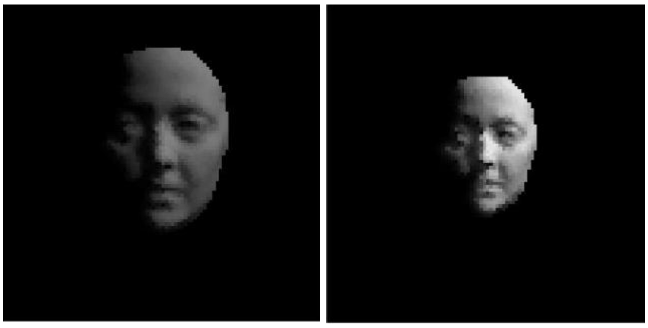


Figure 3 (Left) R1, L2, f = 40mm
(Right) R1, L2, f = 33mm

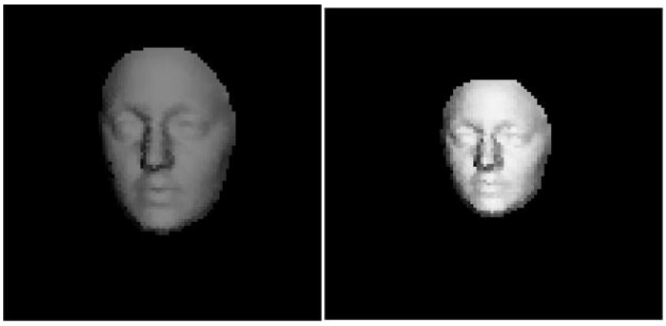


Figure 4(Left) R2, L1, f = 40mm
(Right) R2, L1, f = 33mm

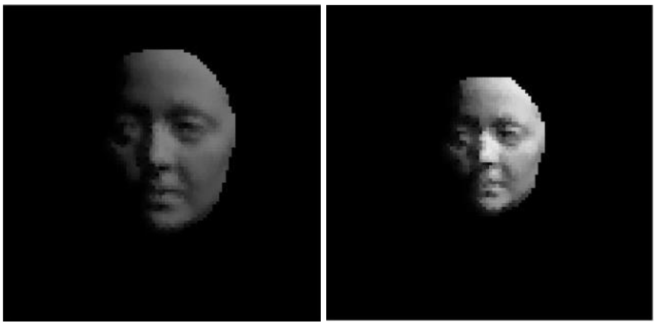


Figure 5(Left) R2, L2, f = 40mm
(Right) R2, L2, f = 33mm

Two images in each figure are presented side by side with different focal lengths. This clearly highlights the effect of the focal length when the other parameters are kept the same. We can see that, with f_2, which is the smaller focal length, the image intensity is considerably brighter and the image size is slightly smaller as compared to f_1. This true for all sets of the images.

We can, therefore conclude that having a smaller focal length generates higher intensity values. This was expected since I varies inversely with f^2 in the radiometric equation. Also, the size of the projected image is expected to be smaller as well because all (c, r) values generated have a positive relation with $f \times S_{x/y}$ and therefore, for the smaller focal length, the resulting values are closer to each other and the image is therefore scaled down.

The result of changing the lighting direction vector is quite visible as well. Compare figure 2 with 3 and figure 4 with 5.

When we change the direction of the light from L_1 to L_2 we see shadows appear on the left side of the face. We can deduce that L_1shines directly on the face from the front where as L_2represents a light source shining from the right. This produces lower intensity values for the left side of the face and bright ones for the right side of the face.

Also, if we notice the intensity values even more carefully, we can tell that the light source for L_2 is on the 'top' right of the face and that it is a little closer to the object than light source corresponding to L_1. This is because the top right half of the face is much brighter than the bottom right half. This can be related to the coordinates of the lighting vector as well where both the x and y coordinates are changed, therefore moving the light source in a diagonal direction. Also, since the norm both lighting vectors is equal to 1, we know that $L_2$is the same distance away as L_1.

Out of the three parameters that were changed, the effects of changing the rotation matrix are the clearest. Changing the rotation matrix from R_1, which is just a 3×3 identity matrix, to R_2 changes the orientation of the camera relative to the face. This is evident from figures 4 and 5 where the face is turned towards the left (looking to its right). Here, we see that some parts of the face on the left are hidden from us and therefore there is a shadow cast upon that area. Note here, that if we keep rotating in the same direction, since we won't any 3D points of the face near the ear region, our model will project an even fewer part of the face and illuminate it.

V. Extra Credit - Results

For extra credit, the projection equation for the weak and orthographic model were implemented as part of the code and the results for all of the scenarios with each model are presented in figure 6, 7, 8 and 9.

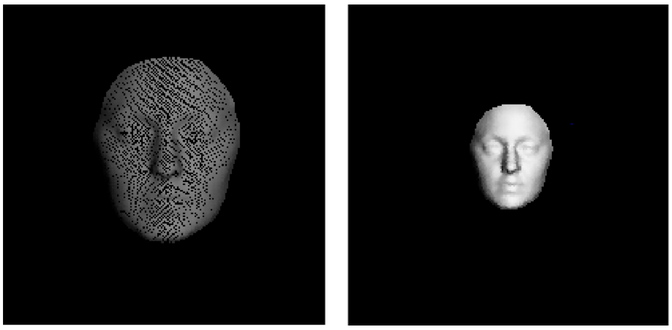


Figure 6. .Results for Weak Perspective Projection
(Top): R1, L1, with both f1 and f2
(Bottom): R1, L2, with both f1 and f2

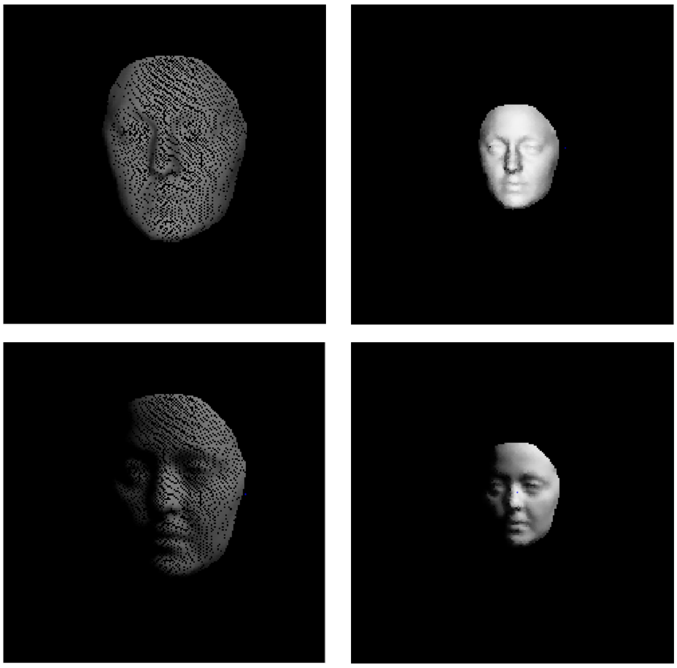
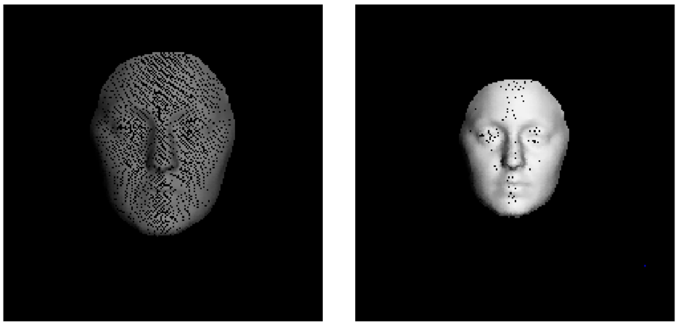


Figure 7. Results for Weak Perspective Projection
(Top): R2, L1, with both f1 and f2
(Bottom): R2, L2, with both f1 and f2



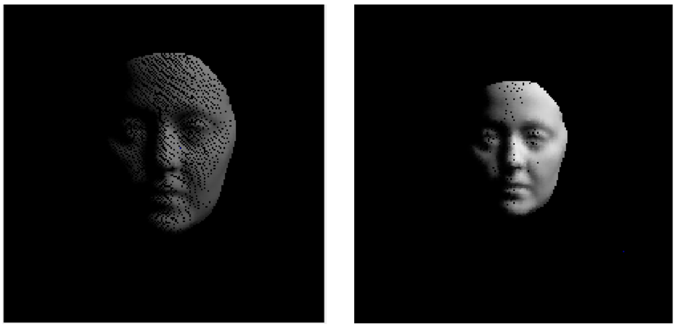


Figure 8. Results for Orthographic Perspective Projection
(1st): R1, L1, with both f1 and f2
(2nd): R1, L2, with both f1 and f2

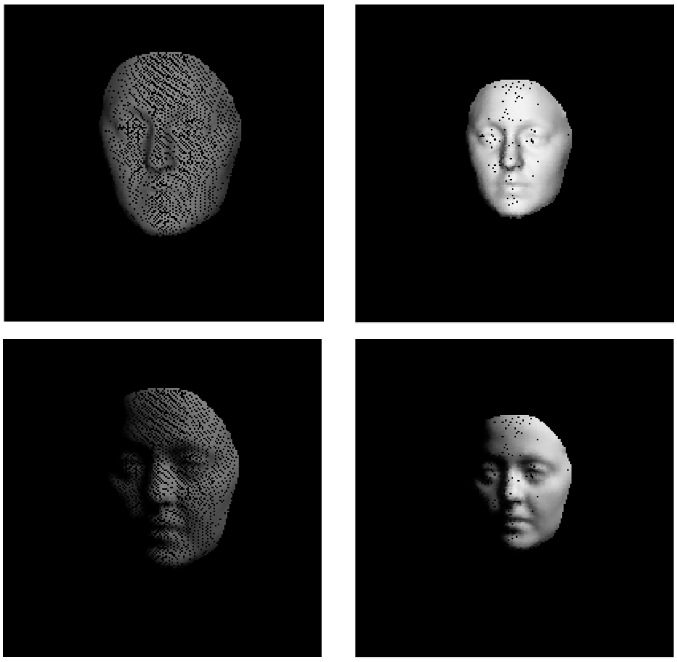


Figure 9. Results for Orthographic Perspective Projection
(Top): R2, L1, with both f1 and f2
(Bottom): R2, L2, with both f1 and f2

The first thing to note about the Weak and Orthographic Perspective Projection results is that the images had to be scaled and shifted to fit inside a 200×200, 2D array. The column and row coordinates I got for Weak Perspective projection were scaled up by a factor of 50 and the coordinates for Orthographic Projection were scaled down by 500.

Before we analyze the results we should notice that Weak Perspective Projection works best if $\delta z_c < \frac{\overline{z_c}}{20}$ and if we look at the variance of Z_c, it comes out to be 340 whereas the mean is about 1000. This means the condition for weak perspective projection is satisfied. But in the results, we still get a lot of black holes and this is because the dividing by $\overline{z_c}$ for all coordinates instead of their own z_c gives a lot of (c, r)locations that overlap that aren't even being filled with averaging later.

A similar case is seen for orthographic projection where the number of black holes for each of the images is higher. (seen easily by comparing Figure 6 to Figure 8). This is seems to make sense since the size of the image is expected to be the same as the size of the object because the projection rays meet at infinity than the lens center. And the density of the 3D points directly affects the resulting 2D image. This also explains why the image coordinates had to be scaled down to fit.

VI. Conclusion

The goals of the project, to apply projection and illumination models were met successfully. Based on the results, we can conclude the effects of the 1) focal length, 2) light direction and 3) the relative rotation between the camera and the 3D object on a projected 2D image.

Specifically, we discussed the Fundamental Radiometric Equation for pixel intensity-calculations and applied three projection models, namely, the Full, Weak and Orthographic Perspective projection models.

We deduced that decreasing the focal length increases intensity for our models and it reduces the size of the 2D image. Furthermore, we determined how lighting direction affects pixel intensities and casts shadows even though the source might be the same distance away in 3D and finally, we established how the rotation matrix that characterizes the relative rotation between the camera and the 3-D object affects which 3D points are projected on the 2D image (and which are not).

Numerous lessons were learnt over the course of the project. Applying what we studied in class, not only reinforced the concepts of projection models but it also helped me to visualize them and therefore gain intuition on their use. Moreover, debugging the code for possible errors not only increased my patience but helped me figure out how the projection matrix was working and understand the units better for each of the parameters. Finally, a lot was learnt while researching on other illumination models and helped me develop a better appreciation for computer vision.

References

[1] Ji, Qiang. "ECSE 6650 Lecture Notes." N.p., n.d. Web. 18 Feb. 2016. <*https://www.ecse.rpi.edu/Homepages/qji/CV/ecse6650_lecture_notes.html*>

[2] FOLEY, James D., Andries VAN DAM, and Steven K. FEINER. "16/Illumination Models and Shading." *Computer Graphics: Principles and Practice*. Reading, MA: Addison-Wesley, 1997. N. pag. Print.