

Handed out: February 12, 2008

Due on: February 14, 2008

Revised Problem Set #1

Cameras and Images Formation

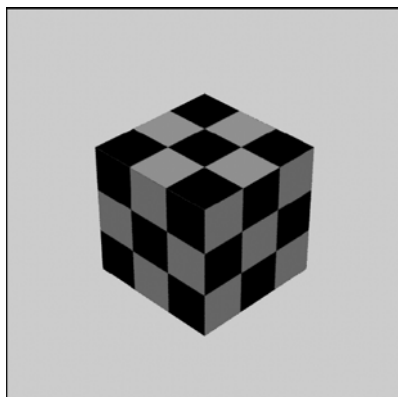
Problem 1-5: Extra Credit

Problem 6: MATLAB

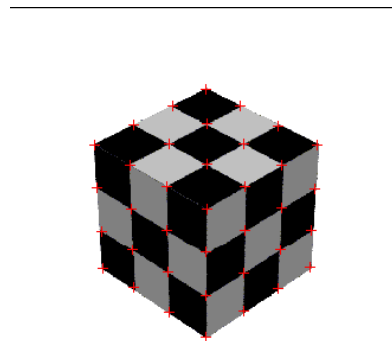
The goal of this problem is to implement a linear calibration algorithm in Matlab based on the method described in section 3.2 (Forsyth). From the following image, we want to extract the intrinsic (focal length and center of image) and extrinsic (rotation and translation) parameters of the camera used to grab this image. We assume no radial distortion.

A usual way to calibrate a camera is to take a picture of a calibration object, find 2D features in the picture and derive the calibration from the 2D features and the corresponding position in 3D. In our case, we use as calibration object a cube 5m wide with a checkerboard pattern on it (see figure 1a). We search in the image (of size 640x640) for the 2D features corresponding to corners of the checkerboard. Figure 1b shows these features.

Since we know its exact size (5m), we can find the exact 3D position of each 2D feature relative to the bottom corner of the cube. This process of finding correspondences is simple but time consuming. We did this part of the work for you. Features2D.mat and Features3D.mat contain the 2D corner features and the corresponding 3D positions.



1(a) Original image



1(b) Detected Features

(a) Your first task is to write a Matlab function which takes these two lists as input and returns the calibration parameters as output. Your function should have the following syntax:

```
function [alpha, beta, theta, f, u0, v0, R, t] = calibrate(f2D, f3D)
```

where α and β are the horizontal and vertical scale factors of the camera CCD (in pixels), θ is the camera skew (in radians), u_0 and v_0 are the center of the image (in pixels), and $[R \ t]$ is the relative rigid transformation between the center of the cube and the camera. Also compute the camera focal length, f , in meters. The size of the camera CCD is (1 x 1) square inches and its pixels are square (i.e. $\alpha = \beta$).

(b) To check your solution create the function,

```
function f2D = pcamview(f3D, alpha, beta, theta, u0, v0, R, t)
```

where f_{2D} are the 2D projections of the points f_{3D} , and the camera intrinsic and extrinsic parameters are as defined above. Using this function project the provided 3D features onto the camera focal plane. Check your solution from part (a) by superimposing the projected points onto the original image checkercube.bmp.

Please submit the files calibration-[last name].m and pcamview-[last name].m by email to cse527_submit@cs.sunysb.edu. In your problem set solution, write the values (α , β , θ , f , u_0 , v_0 , R and t) you found using the cube data set. Also include the result image of part (b) and a printout of your code.

Radiometry and Reflectance Map

Problem 7: Extra Credit

Problem 8:

Based on the photometric stereo method described in section 5.4 (Forsyth), implement a Matlab function which takes images as input and returns the depth, normal and albedo for each point. Your function should have the following structure:

```
function [Nx, Ny, Nz, a] = photostereo(I1, I2, I3, I4, I5, I6)
```

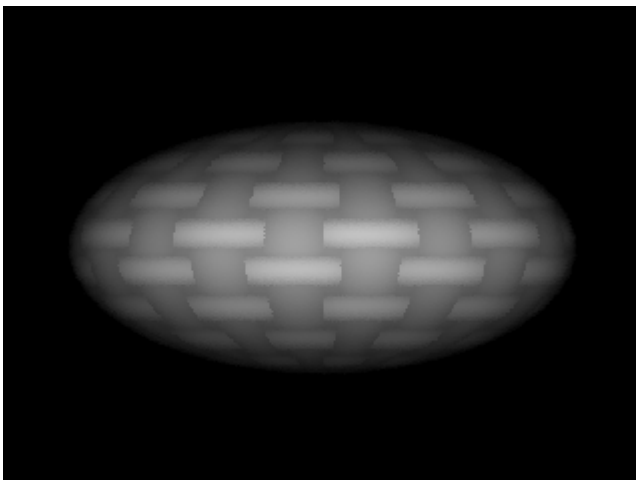
where N_x , N_y and N_z are the components of the normal vector at each pixel, and a is the albedo for each pixel.

To test your algorithm, use the test images `Ellipse1.bmp`, `Ellipse2.bmp`, `Ellipse3.bmp`, `Ellipse4.bmp`, `Ellipse5.bmp` and `Ellipse6.bmp`, which are also shown on the following page. The position of the spotlight for each image is described in the following table:

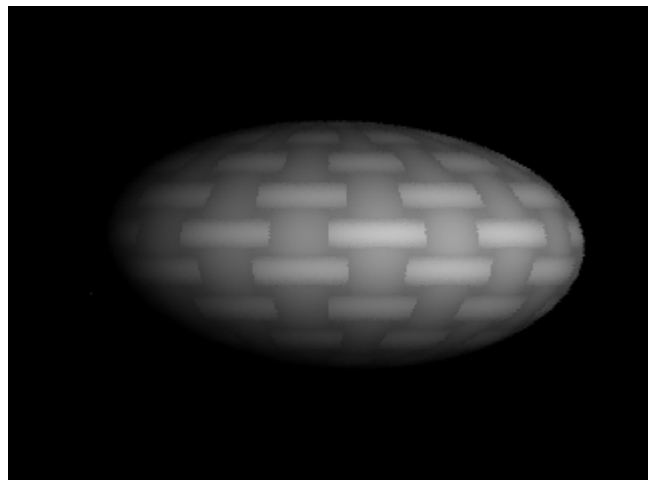
	X	Y	Z
<code>Ellipse1.bmp</code>	0	0	40
<code>Ellipse2.bmp</code>	20	5	40
<code>Ellipse3.bmp</code>	-5	15	40
<code>Ellipse4.bmp</code>	-10	-10	40
<code>Ellipse5.bmp</code>	5	-10	40
<code>Ellipse6.bmp</code>	5	20	40

In this scene the origin is at the center of the ellipsoid, the Y axis points upward, X to the right, and Z out from the screen; orthographic projection is used. The ellipsoid is 2 x 2 x 1 meter high. Spotlight positions in the table above are given in meters.

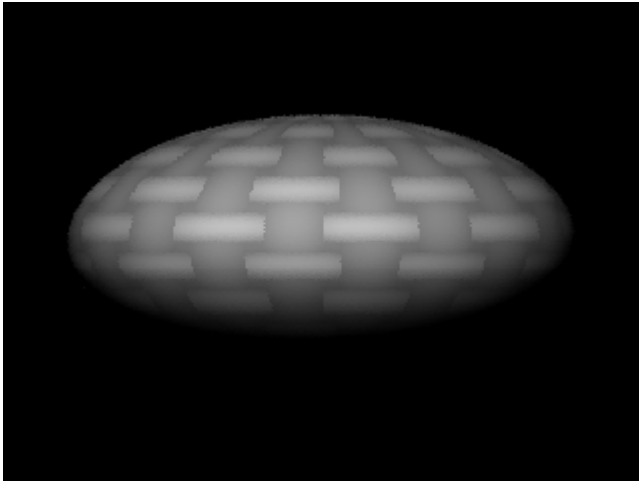
You should submit your file `photostereo-[last_name].m` by email to cse527_submit@cs.sunysb.edu. In your problem set solution, insert a printed copy of the normal vectors graph (using `quiver()`) and the albedo image (using `imshow()`) you found using the ellipse data set.



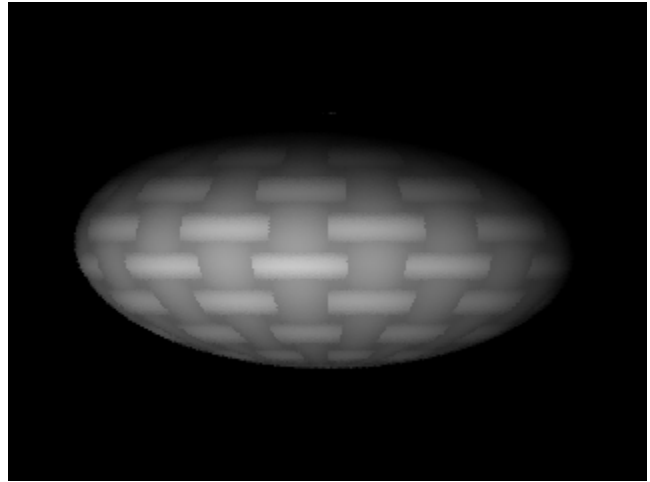
Ellipse1



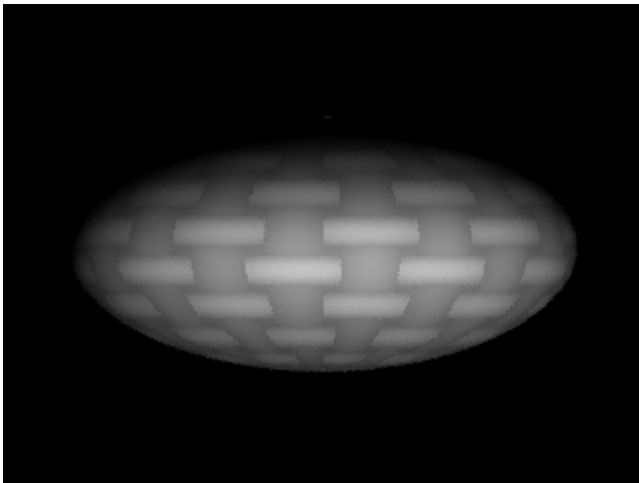
Ellipse2



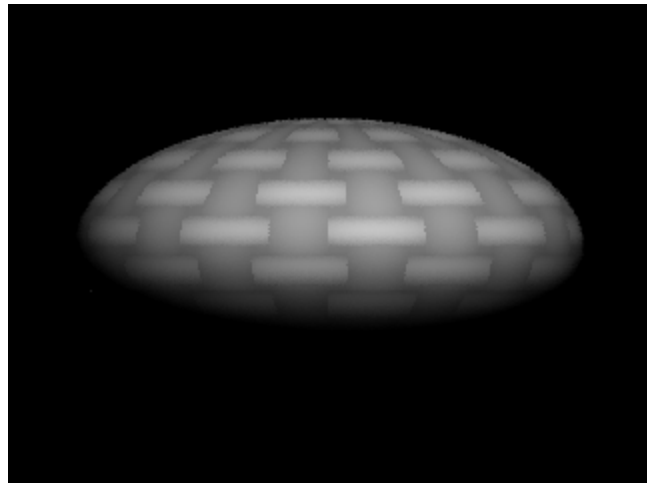
Ellipse3



Ellipse4



Ellipse5



Ellipse6

Problem 9: Extra Credit