BLG634E 3D Vision – Project Part 1

Guideline for submitting your homework:

In submitting your homework:

- Comment your codes clearly.
- All your code should be written in Python language.
- The code of this homework will be the basis fort he final project. Thus, be careful to write your code as functions.
- Do not send your HWs via e-mail. No exception!

Task 1- Plotting a 3D Object

Using numpy-stl, mpl_toolkits and matplotlib libraries read the "cow.stl" and plot the 3D cow as in the Figure 1, using the skeleton code given below.

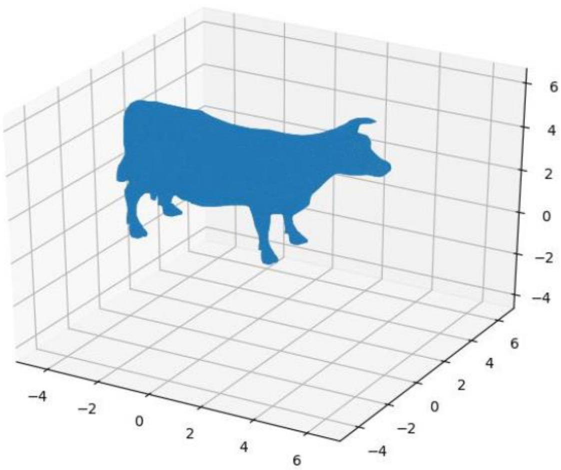

Figure 1: The cow in 3D Grid

```
from stl import mesh
from mpl_toolkits import mplot3d
from matplotlib import pyplot
import numpy as np

figure = pyplot.figure()
axes = mplot3d.Axes3D(figure)
mesh_cow = mesh.Mesh.from_file('cow.stl')
print(mesh_cow.points.shape) #(5804, 9) Each triangular face of the cow
in column view
print(mesh_cow.vectors.shape) #(5804, 3, 3) Each triangular face of the
cow in 3x3 view. Each row represents a vertex.
axes.add_collection3d(mplot3d.art3d.Poly3DCollection(mesh_cow.vectors))
#Add the 3D faces to the created matplotlib axes
min = np.min(mesh_cow.vectors.reshape(-1))
max = np.max(mesh_cow.vectors.reshape(-1)) #Find minimum and maximum
units to place the cow in a cubular grid.
```

```
axes.auto_scale_xyz([min, max], [min, max], [min, max]) pyplot.show()
pyplot.show()
```

Then, use a **homogenous transform matrix** on each vertex to squeeze the model a bit to add a **"calf"** nearby the cow as given in Figure 2.

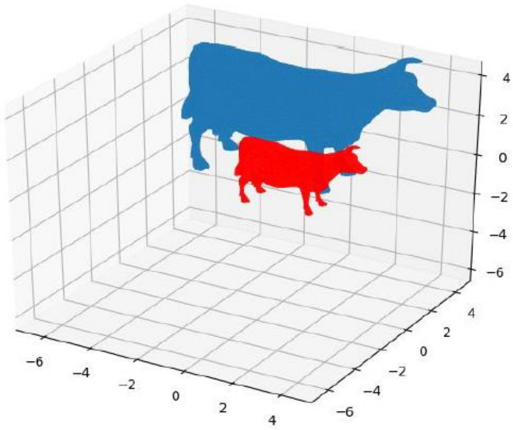

Figure 2:The cow and the calf

Task 2 - Exponential Coordinate Representation of 3D Rotations

Start with a rotation angle of 45 degrees around z-axes. Calculate your 3x3 rotation matrix R_o, using the corresponding w_o vector in R^3 and using Rodrigues formula.

Rotate the cow object by the 3D rotation matrix Ro and display your original object and the rotated object superimposed on each other.

<u>Visualization:</u> Use the rendering given in the previous part with transparency and different colors for each cow as given in Figure 3.

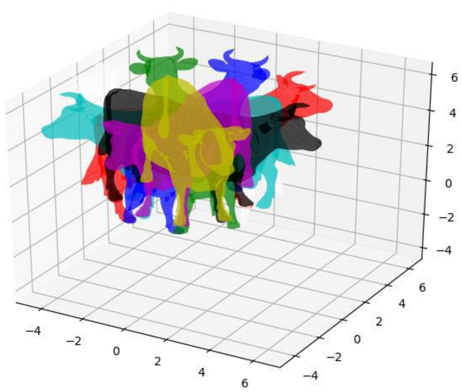

Figure 3: Rotated Cows

(ii) Repeat (i) with a rotation around the axis: w2 = [0.3, 0.7, -1]. Use norm of the axis vector as the amount of rotation. Use default camera view.

Typically, you have to choose the center of rotation appropriately. If the object is already centered, you will not have to worry about that.

Task 3: Tańcząca Polska Krowa

i. Rotate your object around w_1 = [1, 0.5, 0], then rotate the result around w_2 = [0.3, 0.7, -1] vector. Use the norm of each w vector as the amount of radians for rotation. Show your result.

ii. Now rotate your object first around w_2 then around w_1 vector given in (i).

Do you obtain the object in the same position after (i) and (ii) ? Can you say that 3D rotations are commutative or not?

Task 4: Perturbed Rotations

Now, choose an axis vector w, and perturb w vector in each of its 3 components by 2 times (e.g. +- 5). Using Rodrigues formula, calculate your 3x3 rotation matrix R_i for i=1,...,6.

Rotate your 3D object by each of the 3D rotation matrices R_i and display all the rotated objects superimposed with different colors and transparency.

Task 5: Quaternion Representation of 3D Rotations

Carry out the Task 3 by quaternions. Convert those rotations given by their axis vectors to quaternion representations q_1 and q_2. Carry out the consecutive quaternion multiplications to obtain the resulting quaternions and their corresponding rotation matrices. Comment on the result.

Task 6: Explain by a few sentences

I) Is there a singularity in the Exponential Coordinates? If yes, can it be overcome?

II) Is there a singularity in Quaternions?

III) Why is the quaternion representation preferred over Euler angle representation for 3D rotations?

Task 7: Computing Angles

Read Dr. Gregory G. Slabaugh's report titled "Computing Euler angles from a rotation matrix"[1]. Find which Euler angles are used in the transformation matrix given below. Apply the rotation to the object

I. Directly
II. Using rotation matrices Rx, Ry, Rz.

and show that your findings are true.

---

[1] Slabaugh, G. G. (1999). Computing Euler angles from a rotation matrix. Retrieved on August, 6(2000), 39-63.

$$\begin{bmatrix} 0.8365 & -0.5245 & -0.1585 \\ 0.2241 & 0.5915 & -0.7745 \\ 0.5000 & 0.6124 & 0.6124 \end{bmatrix}$$

Task 8: “Kobieta, wołasz mnie”

The first time I’ve listened Myslovitz’s self-titled album, every song in it stuck in my head for at least a week. I was just like the cat in this(https://web.itu.edu.tr/sahinyu/hw2_cat.mp4) video. In the first task of the homework, you will create a video for your favourite album. A frame of the target video is given in Figure 4.

Figure 4: The cat is wishing to get rid of the album, but it is impossible.

- Create a572x322 pixels sized empty image.
- For every frame of the video I extracted the corner points of each plane. Use your favourite album cover to change the video. For each plane’s correspondences, create transform matrices (**Do not use OpenCV’s built-in functions. Find coefficients using $\boldsymbol{q}=\boldsymbol{M}^{-1}\boldsymbol{A}$.**)
- Warp the album covers using matrix multiplication and place onto empty image.
- Complete the video by adding the cat. Decide which planes are behind or front of the cat considering areas of the planes.
- While my augmented image looks pretty good, there may be a pixelation effect on the edges of the album cover, making the image look somewhat synthetically generated. How might this be improved? Explain in few sentences.

Task 9: Pier 69

In this task, you will implement and experiment with estimation of a Homography matrix between images of a given scene (planar scene) from different views. You will use the Normalized Direct Linear Transform with SVD we studied in the class to estimate the 3x3 Homography matrix H. You will then use the estimated transformation to form a panoramic image, i.e. a combination of the three images by simple stitching.

In this task, you will select the corresponding points (x,y) manuallly. You can use Microsoft Paint's pixel picker tool or any other similar one.

Load the pier images given in Figure 5. Try to choose rich interest points such as corners that are visible in both images. What is the minimum number of corresponding feature points to be selected from the images? Due to noise in measurements, you should try to choose more than 4, e.g. I chose 6 or 7 points for each image.

Figure 5: Three different scenes from the pier.

Using the set of feature point coordinates from the images, write a functiton **homography2D** to estimate the Homography matrix H using the normalized direct linear transform that employs SVD for solution of the homogeneous linear system we learned in class. You should normalize each set of points separately, which makes the origin at the centroid of all the points, and mean distance from the origin is set to sqrt(2). You should write another **normalise2Ddpts** function. Then write a routine to transfer coordinate points in one image to another using the estimated homography matrix.

You can find the bounding box of the panoramic image onto which you can overlay both of the images. Use the corners of the input image to transform the image bounding box to the output domain and find the coordinates of the big bounding box (panoramic coordinates) that includes both image domains. Transfer these coordinates to the input image domain to resample your input image. Then place your output image over the panoramic image domain.

Rather than copying the second image over the first one as you did before, try simple image blending ideas, such as taking convex combinations of images etc. How does the result differ from that of simple copying? Print/display your result. An expected result is given in Figure 6.

Figure 6: An expected output.

Task 10: SIFT & RANSAC

In this task, you will use a feature extraction and matching technique based on a technique such as SIFT to automatically extract a set of corresponding point coordinates from the given images. You can use a built-in function or already available routine from another library to do this task. Using the extracted feature point sets, estimate the Homography matrix.

If you're not satisfied with both of the results, what are possible reasons?

Next, use RANSAC method we learned in class to eliminate outlier point correspondences. Play and experiment with different parameters of RANSAC, particularly the support distance t. Then obtain the Homography matrix with a good inlier point correspondence set. Now, stitch the images together automatically. Show your results.

Task 11: CVPR Image Matching Challenge

In addition to the given Pier images, select two images from CVPR Image Matching Challenge[2] and repeat Task 10. How are the results compared to the previous ones? Could you improve the results using some preprocessing?

[2] https://image-matching-workshop.github.io/