

Architectural Geometry – Worksheet 2

This worksheet is intended to help us understand (a) homogeneous coordinates, (b) rigid body motion, and (c) scaling.

You are allowed (and indeed encouraged) to discuss the tasks with other participants, but you are not allowed to hand in solutions produced by somebody else. If you get stuck, please do not hesitate to contact me, and I will do my best to help you get past the obstacle.

Write up your solutions neatly in a document, and send it together with an archive (e.g. zipped) containing your code to me at `marten.wadenback@chalmers.se` at the latest on the 30th of May.

Task 1: Study the section about homogeneous coordinates on p. 203. Create a function `c2h` which turns a list of Cartesian coordinates into homogeneous coordinates, and a function `h2c` which transforms back from homogeneous coordinates to Cartesian coordinates. Hint: the `repmat` function may be useful here. The functions `c2h` and `h2c` should behave as:

```
>> x = randn(2,6)

x =

    1.0933   -0.8637   -1.2141   -0.0068   -0.7697   -0.2256
    1.1093    0.0774   -1.1135    1.5326    0.3714    1.1174

>> X = c2h(x)

X =

    1.0933   -0.8637   -1.2141   -0.0068   -0.7697   -0.2256
    1.1093    0.0774   -1.1135    1.5326    0.3714    1.1174
    1.0000    1.0000    1.0000    1.0000    1.0000    1.0000

>> h2c(X)

ans =

    1.0933   -0.8637   -1.2141   -0.0068   -0.7697   -0.2256
    1.1093    0.0774   -1.1135    1.5326    0.3714    1.1174

>> h2c(2*X)

ans =

    1.0933   -0.8637   -1.2141   -0.0068   -0.7697   -0.2256
    1.1093    0.0774   -1.1135    1.5326    0.3714    1.1174
```

Task 2: Let us represent points using homogeneous coordinates with last coordinate equal to one, and vectors using homogeneous coordinates with last coordinate equal to zero. Using the familiar operations in \mathbb{R}^n , verify that

(a) $[\text{vector}] + [\text{vector}] = [\text{vector}]$

(b) $[\text{vector}] - [\text{vector}] = [\text{vector}]$

(c) $[\text{point}] - [\text{point}] = [\text{vector}]$

(d) $[\text{point}] + [\text{vector}] = [\text{point}]$.

What happens if we try to compute $[\text{point}] + [\text{point}]$?

Task 3: Explain what the rigid body motions represented by the G and H do independently, where G and H are given by

$$G = \begin{bmatrix} 1 & 0 & 0 & -3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

What transformations do GH and HG represent? The inverses of G and H can be found almost without computing anything – explain how and why!

Task 4: To apply a rigid body motion to a list of vertices, we need to perform the steps (these are done in `renderObject.m`):

Step 1: represent the rigid body motion as a matrix,

Step 2: represent the vertices using homogeneous coordinates,

Step 3: multiply them by a matrix representing the rigid body motion, and

Step 4: go back to Cartesian coordinates.

- (a) The function `scaleObject` is meant to apply a scaling transformation to an object, but currently does nothing. Study `scaleObject.m` and think about lines 15 and 16. Which of them is correct, and why? Uncomment the correct one.

Sample usage of the `scaleObject` function:

```
T = scaleObject(cube,[0.5 2 1]');
renderObject(T);
```

- (b) Use the skeleton file `translateObject.m` to make a function that applies a translation to an object. Verify that it works, e.g. with the commands:

```
T = tetrahedron;
renderObject(T);
hold on;
T2 = colorObject(T2,'green');
T2 = translateObject(T,[2 3 4]');
renderObject(T2);
```

- (c) Use the provided function `rotm` (or `rotmd`, if you prefer degrees instead of radians) to create a corresponding function `rotateObject` which rotates an object the angle φ around the vector v . The command `obj = rotateObject(obj,v,phi)` should give the rotated object.
- (d) Suppose we need to scale, translate, and rotate an object. What is the most natural order to perform these operations?