T5 answer sheet

1a. Method: calculate view vector (T-P) and normalize it; this is the z-axis of the coordinate system. Calculate vector perpendicular to view and up using the cross product and normalize; this is the x-axis of the coordinate system. Finally calculate the y axis as the cross product between x and z. Fill in the vectors in a 4x4 matrix, using the camera position as translation (column 4).

b. Using the result from a: the centre C of the screen is at P+z; then the corners are at C-x-y, C+x-y, C+x+y and C-x+y if we ignore aspect ratio; scale y by 768/1024 to get the correct aspect ratio.

c. This is screen centre C.

d. Using the results from a and b, calculate the stepsize for one pixel along the x-axis as dx=2x/1024 and the stepsize for one pixel along the y-axis as dy=2y/768. Now, a pixel u,v is at (C-x-y)+u*dx+v*dy.

2a. In this case the rays are all parallel; generate ray targets T on the screen plane as before, but calculate the origin as T-V. The 'camera' thus becomes a plane with the same size as the screen plane.

b. Perspective is a byproduct of using a pinhole camera and a screen plane with a size > 0.

c. Use a curved screen plane.

d. This is left as an exercise to the reader, but should be clear at this point. ☺

3. Ray coherence refers to rays travelling in a similar direction, so that a bundle of rays forms a tight beam. As a consequence, the rays traverse the same scene regions, which can be exploited to speed up BVH traversal.

4. We may find intersections behind the camera.

5a. We transform the ray with the inverse matrix of M1 (instead of transforming the object O) and test for ray/object intersection. Then, we transform the (original) ray with the inverse matrix of M2 and test for ray/object intersection again. Note that the object is never transformed.

b. The normal that we find is in object space and must be transformed to world space using the appropriate matrix (M1 or M2).

c. Matrix M1 and M2 may include a scale factor, in that case the ray must use the inverse of this scale to find the correct distance along the ray during intersection. Normalization would destroy this scale.

6. Draw the situation to quickly see the answer. This process is referred to as CSG (constructive solid geometry).

7. See book, page 238 (and particularly figure 10.7). Only difference is the direction of d, which is inverted here.

8a. A spatial subdivision cuts up space, which may lead to the same object overlapping multiple areas (e.g., in a grid). An object subdivision cuts up a list of objects (e.g., in a BVH), which may lead to overlapping spaces.

b. It is a spatial subdivision.

c. It is still a spatial subdivision, although it reduces the number of cases where objects end up in multiple subspaces.

d. A kNN search is usually applied to points; in that case one could argue that a spatial subdivision is the same as an object partition.

9. This was not covered in the lecture (probably, writing this before the lecture), but:

The cost function multiplies primitive count by surface area for each bounding volume. This ignores the fact that primitives may occlude each other inside the volume. Some primitives may not by reachable at all, if they are enclosed by other primitives.

Also, the SAH cost function assumes ‘random rays’; when the ray distribution is not random at all (e.g., when sampling a distant area light in a distributed ray tracer, where many shadow rays have a similar direction) this is inaccurate and potentially not optimal.