Homework #1

Perception, Graphics Systems, Filtering,

Transformations, Hierarchies

Prepared by: David Doan and Matthew Milcic

Assigned: Monday, October 11th
Due: Monday, October 25th

Directions: Please provide short written answers to the questions in the space provided. If you require extra space, you may staple additional pages to the back of your assignment. Feel free to discuss the problems with classmates, but please ***answer the questions on your own.***

Name:____

Problem 1: Color (6 points)

Two geometrically identical objects are in a dark room, sitting on a table across from you. Shining a tungsten flashlight at the two objects, you notice that they appear to have the same color.

(a) (2 points) Is it possible that the light being reflected from the two objects have different spectral power distributions (SPD) even though they look the same? Explain.

(b) (2 points) Attempting to verify the colors of the objects, you think about trying your flashlight with a halogen bulb. Is it possible that under different lighting, the objects would appear to have a different color? Why?

(c) (2 points) You can't find the other flashlight, and this object color mystery continues to drive you nuts. Luckily, you recall something in class about color filters, and realize that your spiffy orange sunglasses are in your book bag. With the sunglasses on, and using the original tungsten flashlight, is it possible for the two objects differ in color? Why?

Problem 2: Display Devices (6 points)

Good news! Your Compaq 12", DSM-GREEN (monochrome) monitor has just arrived! You can finally install Windows 3.1 on your IBM 286. Unfortunately, you find that your monitor flickers when running Wolfenstein 3D.

(a) (2 points) Suggest a change to the display system to help reduce the flickering. This may be a change to the software, display drivers, or even a change to the way the monitor works.

Now that you have solved your flickering problem, you start to realize that neon green isn't your favorite color. Fortunately, your friend wants to get rid of his VGA monitor and doesn't want to pay the $10 recycling fee. While the VGA monitor is better, you don't want to use it if the flickering returns.

(b) (2 points) Will your solution from part (a) still apply to this new monitor? Why or why not?

You finally remember where you put that 19" 1600x1200 monitor that you received for high school graduation, but couldn't fit in your dorm room. After hooking it up, you download the newest patches for Wolfenstein 3D, which promise state of the art 24-bit true-color output for the most realistic first-person shooter experience ever.

(c) (2 points) Will the 24-bit true-color display be able to produce all colors that can possibly be perceived by the human visual system? Explain.

Problem 3: Convolution Filters (6 points)

In this problem, you need to design convolution filters to accomplish several different tasks. For each sub-problem, provide a filter that should accomplish the task. *Justify your choice of filter.*

(a) (2 points) The image you're editing is too dark, and you decide you need to amplify the value of each pixel by a factor of 1.5. Suggest a convolution filter that would augment the value at each pixel of the image without changing it in any other way. (Technically, after scaling pixel values, they could be out of range; assume that any needed clamping will be taken care of later, after filtering).

(b) (2 points) While taking a photograph with you digital camera, you fail to hold the camera steady, and it translates upward while the shutter is open. You discover this later when you see that vertical edges, in particular, have been blurred a bit. You decide to filter the image so that vertical edges are sharpened, but horizontal edges are unchanged. Suggest a single convolution filter that does this.

(c) (2 points) In this problem describe the effects of the following filters. Also which filter will be the least blurred and which will produce the darkest image? *Again, justify your answers.*

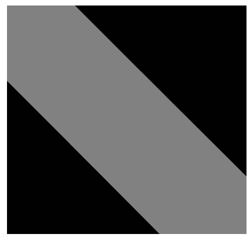
0	-1	0
-1	5	-1
0	-1	0

0	0	0
0	1	0
0	0	0

.3	0	0
0	.3	0
0	0	.3

Problem 4: Image Boundaries (8 points)

When applying a convolution filter, there are many possibilities for handling pixels near image boundaries. For each sub-problem below, discuss the consequences of the given boundary filtering approach when applying a mean filter the following simple image:



Assume that the "right" answer would correspond to the image continuing naturally beyond the boundaries so that the extended image plane would be a gray stripe in a black plane as suggested by the input image. Will a given approach below give the "right" answer all along the boundary? If not where does it give the "wrong" answer and in what way?

(a) (2 points) Calculate only the values of the pixels in the resulting image for which the support of the mean filter is entirely contained within the original image.

(b) (2 points) Pad the edges of the original image with zeros before filtering.

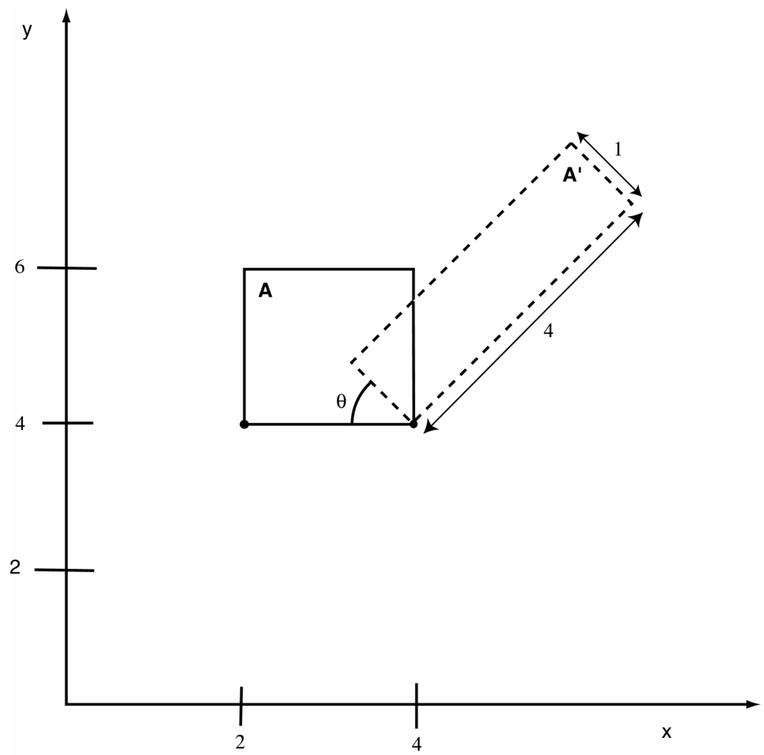
Problem 4: Image Boundaries (continued)

(c) (2 points) Reflect the original image across each image boundary before filtering. You can think of this as reflecting the image across the top, bottom, left, and right boundaries, and then, to fill the diagonal regions near the corners, you would reflect the top reflection to the left and right, and the bottom reflection to the left and right.

(d) (2 points) Perform a "toroidal wrap" before filtering. This kind of wrapping maps the original image onto a torus so that the left edge meets the right edge and the top edge meets the bottom edge. It is equivalent to tiling the plane with the original image.

Problem 5: 2D Affine Transformations (5 points)

Write out a sequence of 3×3 translation, scaling, and rotation matrices that transforms the square **A** into the rotated rectangle **A'**, as shown in the diagram below. Assume that the product **M** of your sequence of matrices will be applied to each point **p** of **A** by post-multiplication, i.e., **Mp** = **p'**. θ is the only variable, all other entries of your matrices should be filled in with numbers.

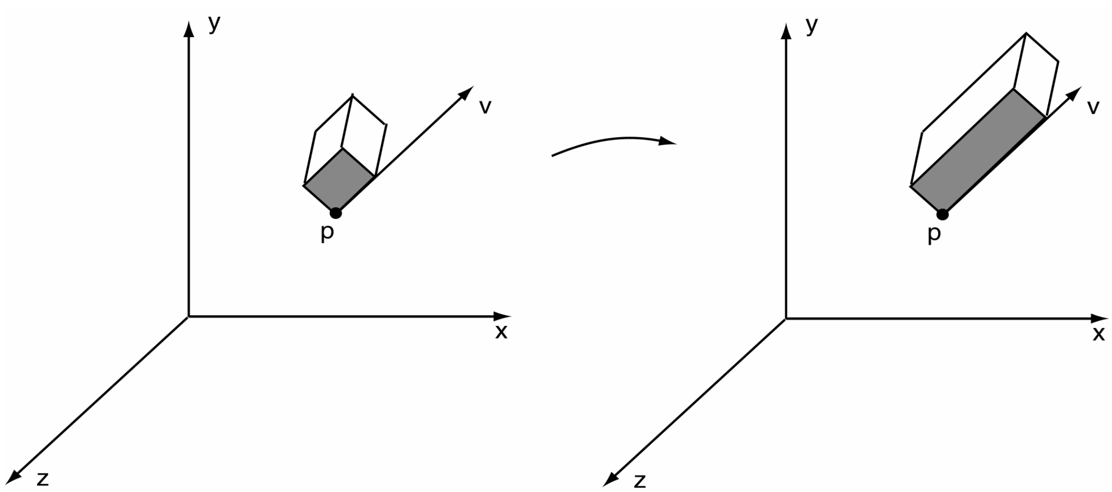


Problem 6: 3D Affine Transformations (5 points)

The basic scaling matrix discussed in lecture scales only with respect to the x, y, and/or z axes. Using the basic translation, scaling, and rotation matrices, specify how to build a transformation matrix that scales along any ray in 3D space. This new transformation is described by the ray origin $\mathbf{p} = (x_0, y_0, z_0)$ and unit direction vector $\mathbf{v} = (x_1, y_1, z_1)$, and the amount of scaling s_{pv}.

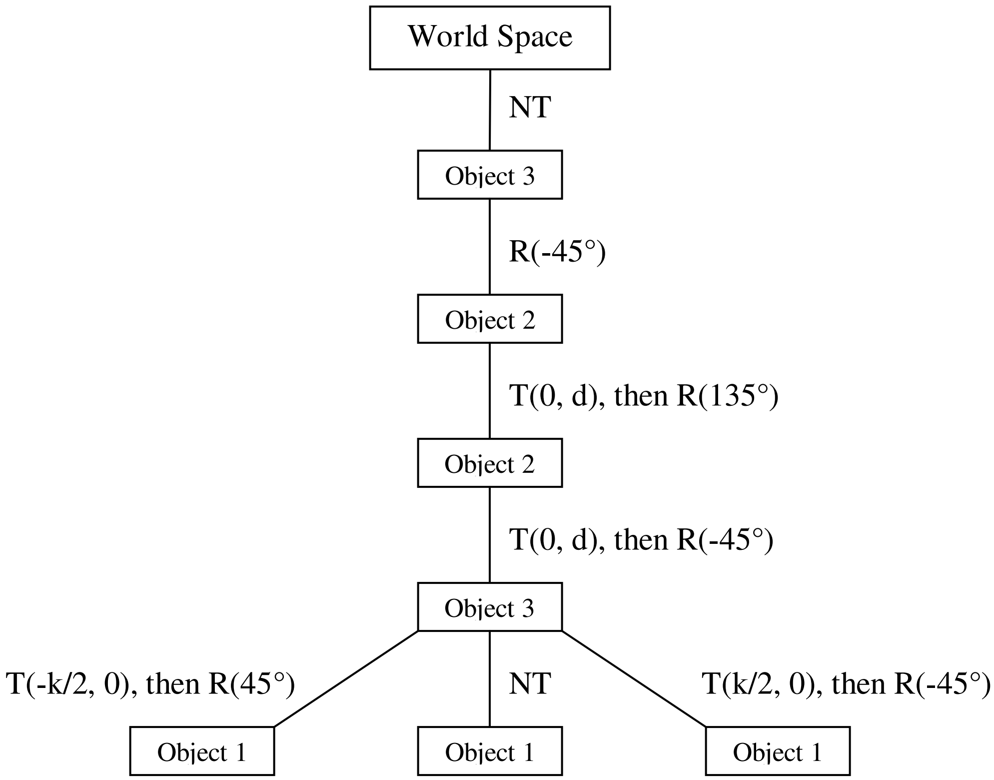
You can use any of the following standard matrices (from lecture) as building blocks: Euler angle rotations $R_x(\theta)$, $R_y(\theta)$, $R_z(\theta)$, scales $S(s_x,s_y,s_z)$, and translations $T(t_x,t_y,t_z)$. You don't need to compute exact formulas for the rotation angles, but you must describe how to compute each of the rotation angles using words and drawings. You don't need to write out the entries of the 4x4 matrices, it is sufficient to use the symbols given above.

For clarity, a diagram has been provided, showing a box being scaled an unknown amount with respect to a given ray. Your answer should work for any ray, not just the case shown in the picture.

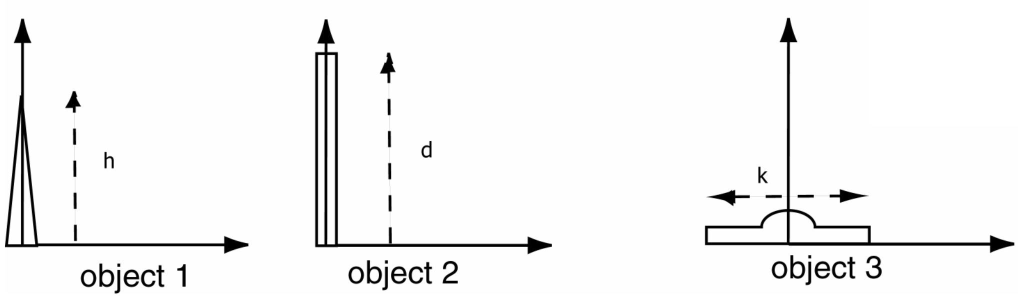


Problem 7: Hierarchical Modeling (5 points)

Suppose you are a graphics system, and you need to render a model from the following tree, with transformations labeled.



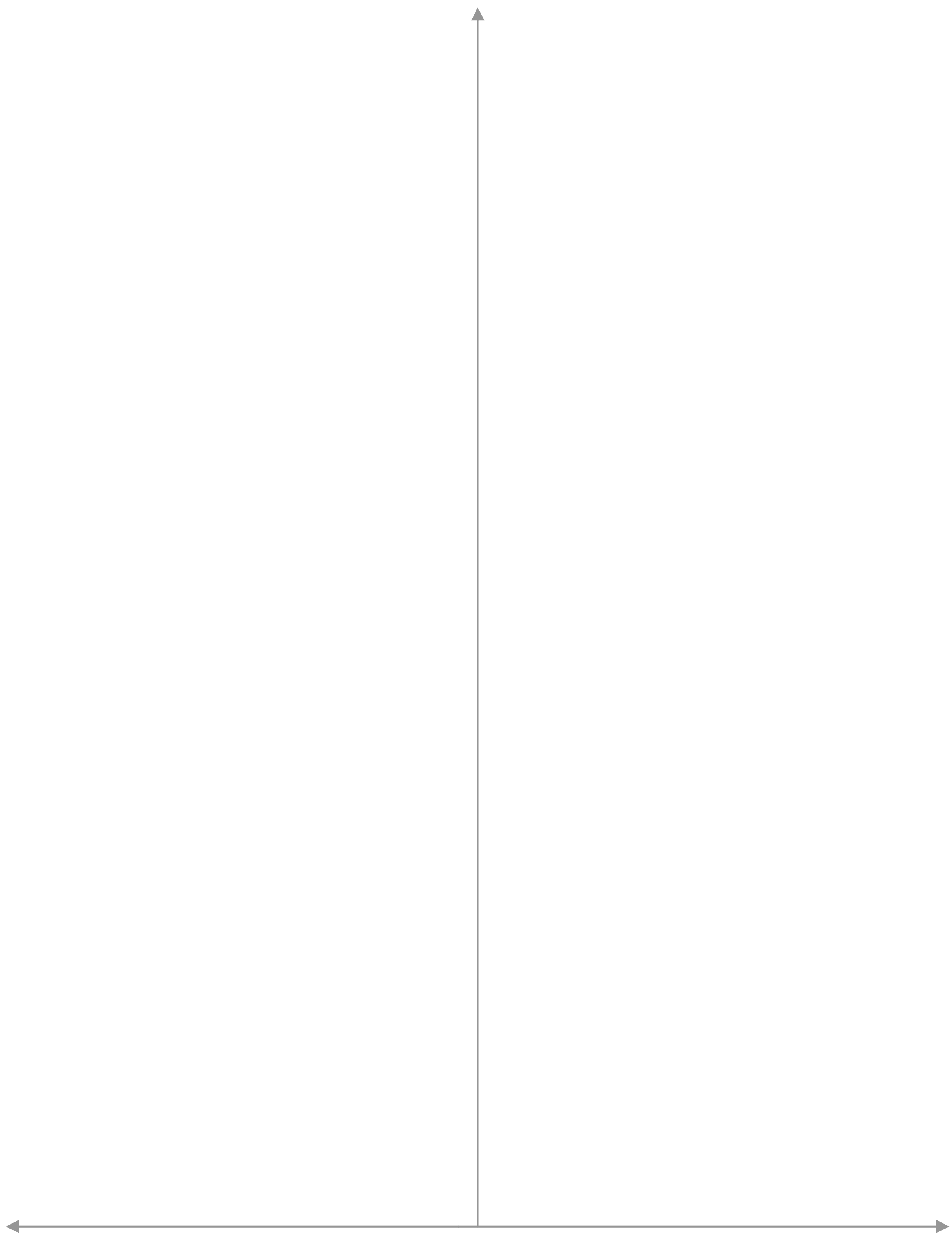
In memory, you find the following primitives:



From your instruction set, you recall that R(θ) means to rotate the primitive by θ degrees (counter-clockwise) about the origin, and T(j, k) means to translate the primitive j units along the x-axis and k units along the y-axis. NT means no transformation.

Problem 7: Hierarchical Modeling (continued)

Render the model using your favorite drawing utensil. Being the organic hardware that you are, your user understands if your model isn't perfect.



Problem 8: Modeler (5 points)

In this problem you will be working with 'your model'. Your model is the model you plan to build for the Modeler project, a simplified version of that model, or a completely different model that *could* be a built for the Modeler project. Your model must have at least four articulated levels of hierarchy. That is, you need at least three joints. The transformation that maps the model into world coordinates (e.g., the translation of the root node) does not count as a joint. The model in the previous problem qualifies as having enough hierarchical levels, but you should not, of course, use that model in this problem (it's not a very interesting model, anyway!). You may work with your partner to design the model and hierarchy you use for your answer.

Sketch your model in two poses and draw the corresponding hierarchy tree. One pose should be the model in a neutral state and one should have changes at each of four levels of your hierarchy along exactly one descending path through the tree. Label all the parts of the model in the neutral pose and the corresponding nodes in the tree. Label the transformations only along the edges that correspond to what changes in moving from the neutral pose to the other pose you provided. You may use generic names for the transformations, as was done in the lecture; e.g., M_{ua} was the "upper arm" transformation for the humanoid figure. You do not need to have separate sketches for each primitive of your model.