NetID:

Computer Science 426 Exam 1 – Spring 2015

This test has 6 questions on pages numbered 2 through 7. Do all of your work on these pages, giving the answer in the space provided (use the back for scratch space or overflow if needed). This is a closed-book exam, but you may use one page (both sides) of notes during the exam.

1. Please write your NetID in the upper right corner of every page.
2. Please write out and sign the Honor Code pledge before turning in the exam.

"I pledge my honor that I have not violated the Honor Code during this examination."

Student Name:

Honor Code Pledge:

Signature:

Question	Score
1	
2	
3	
4	
5	
6	
Total	

1. Colors

(a) 5pt. Two different light spectra that are perceived as the same color by the eye are called “metamers.” Explain briefly: how can this happen?

Apectrum has infinite dimension, and is “projected” to 3 dimensions in the eye, so there are many spectra that must project to the same perceived color.

(b) 5pt. A slice of XYZ space is often depicted as a “horseshoe” shape in the CIE chromaticity diagram. What colors lie along the flat part of the horseshoe and describe briefly how are they different from the colors along the curved part?

Purple colors along that flat edge are not “pure” spectral colors (single wavelength) as on the curved part.

(c) 5pt. What are the “shapes” of the HSL and HSV color spaces? What is the practical difference?

HSV is a hexacone shape, whereas HSL is like two of hexacones sharing the flat hexagon. HSL gives actual lightness so could be used for example to extract the grayscale image.

2. Image processing

Suppose you convolve an image with a Gaussian filter kernel of standard deviation σ to blur the image.

(a) 3pt. What is the mathematically ideal size and shape for the filter kernel?

infinite

(b) 3pt. What is a reasonable size and shape used for practical image processing?
What is the difference in the result, relative to that of (a)?

2σ or 3σ

(c) 3pt. How do you handle the edge cases where the kernel extends outside the image?

One option: omit those samples but make sure that remaining filter weights sum to 1.0.

(d) 3pt. What is the difference between the Gaussian filter and the Bilateral filter?
Why might someone prefer the latter?

Gaussian weights fall off with spatial distance; weights in the bilateral filter are the product of Gaussian terms for spatial distance and color distance. Bilateral smoothing is good for preserving contrast across visual edges in the image.

(e) 3pt. Could you combine use of a 2 x 2 ordered dither matrix with Floyd-Steinberg error diffusion?
Explain briefly: If not why not? If so, how would it work?

Yes. A 2x2 dither only allows for 5 possible output colors. So to achieve more shades of gray, you could do the dither, and after deciding the color each pixel propagate the error incurred to neighbors using F-S.

(f) 3pt. When warping and image, what are the tradeoffs between forward mapping and reverse mapping?

Forward mapping uses the natural mathematical function of the warp but might leave gaps in the result (and can fold over itself). Reverse requires inverting that function (which may not be easy or even possible) but covers every output pixel exactly once.

3. Compositing

(a) 3pt. What are two possible physical interpretations of the alpha channel when compositing images using the "over" operator?

1. Transparency (like glass)
2. Partial coverage (like hair at the edge of a mask for a person's head).

(b) 3pt. Provide a high-level description of the goal function and constraints used in Poisson image blending?

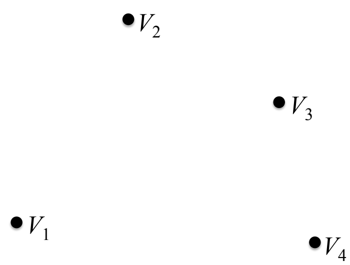
Match the gradients in the pasted region to those of foreground image, subject to boundary conditions on the background image at the edge of the mask region.

(c) 3pt. In what situation would you benefit of using Poisson image blending rather than simple alpha blending, and why?

If the overall color statistics of the foreground image are not well matched to those of the background, for example imagery with pasting greenish water into a bluish water scene.

NetID:

4. Parametric Curves



(a) 4pt. In the space above, draw the cubic Bezier curve for the four control points shown above.

(b) 4pt. Also draw and label four control points V5,V6,V7,V8 for a cubic Bezier curve that joins the curve sketched in (a) with C^1 continuity.

(c) 4pt. Suppose these two Bezier curves were two adjacent sections of a cubic B-spline. Draw and label five control vertices B1...B5 that could plausibly be the control points associated with these two sections.

(d) 4pt. The joint in (c) has C^2 continuity. Briefly: how is that different from G^2 continuity?

Overall notes: Needs drawings. First bezier should be contained within convex hull of V1..V4. The curve should end at V1 and V4 and should be tangent to the first and last edges of the polygon at those points. V5 = V4, and V3-V4 should be colinear with V5-V6. The first segment (Bezier curve) should be near B2 and B3, while the second should be near B3 and B4. Both segments should be inside the convex hull of B1…B5. C2 continuity means the curve is twice differentiable while G2 means that the tangent and curvature (osculating circle) of the shape in the plane should be continuous.

5. Surfaces

(a) 3pt. What are three advantages of using an implicit surface representation over a polygon mesh?

Implicit surfaces advantages : in/out test, concise, guaranteed validity

(b) 3pt. Describe how would you determine if a point P is inside or outside a shape S represented by a BSP tree?

Key concepts that we were looking for include recursive traversal, reject if outside, accept if reaches down the leaf, passing all the tests

(c) 4pt. The "marching cubes" method for isosurface extraction in 3D reduces the $2^8=256$ possible combinations of inside vs outside at the 8 corners of the voxel to only 15 cases (due to symmetries). The 2D analogue "marching squares" for extracting isocontours should have $2^4=16$ cases. Draw the cases here. You may draw all 16 cases or draw fewer due to symmetries; but if you group by symmetries, say how many of the 16 overall cases belong to each one that you draw.

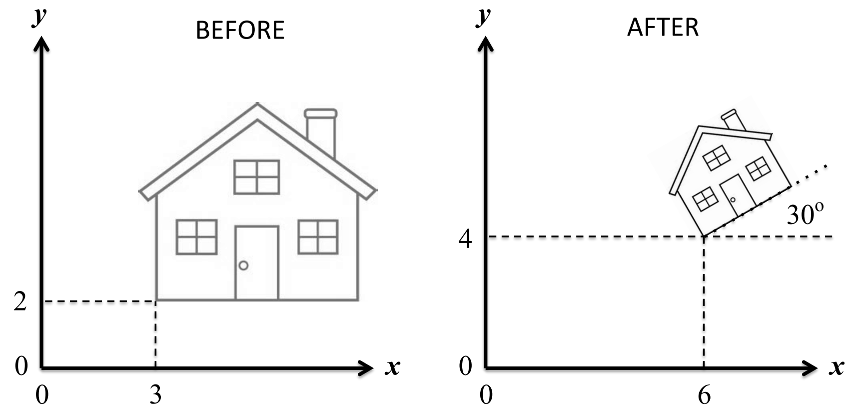
(need pics)

(d) 4pt. Draw the shape that would be extracted by marching lines for the "zero set" (value = 0) in the grid below. Your drawing does not need to be perfect, but should have the correct overall shape and characteristics; try to get it roughly within 0.2 of the true answer. (Suggestion: sketch in light pencil first.)

2.3	1.7	0.9	0.2	0.4	1.7	2.5	3.1
1.2	0.4	0.1	-0.8	-0.1	0.4	1.8	-2.6
0.3	-0.5	-0.7	-1.4	-0.3	0.5	1.3	2.3
0.2	-0.9	-1.7	-2.1	-1.0	-0.1	0.8	1.5
0.2	-0.9	-1.8	-1.7	-0.3	0.7	1.2	1.4
1.2	0.4	-0.6	-0.2	-0.1	-0.4	-0.3	0.8
1.8	1.0	0.7	0.4	1.3	0.5	-0.2	0.8
2.6	1.9	1.7	2.0	2.1	0.9	0.7	1.2

overall criteria were:
- Correct overall shape - 2pt
- Marching squares returns straight lines, not smoothed ones - 1pt
- Correct offsets at edges - 1pt
- Catching the bug in the drawing (-2.6 near upper right) - 1pt bonus

6. Transforms



(a) 6pt. Write the product of matrices that transforms the house shape in 2D as shown above, where after the transformation the house is half as big. You do not need to work out all the math – just write all the components in the right order. However, to ensure partial credit, label the function of each matrix you use.

Correct transformations (Translation to origin, scale, rotate, translation to desired point) -2pt
Correct order - 2pt
Correct matrices - 2pt

(b) 3pt. In a typical graphics library this transformation would be pre-multiplied before applying it to the vertices representing the house. Briefly, why?

Done for efficiency, because the resulting matrix will encode same transformation regardless of what point it is applied to. So multiply all the matrices once to get one matrix, then apply that to all points.

(c) 3pt. Briefly: why do we represent points using homogeneous coordinates?

We use homogenous coordinates to represent larger family of transformations, in particular including translation.