Graphics Qualifier

Spring 2017

Please point out any ambiguities that you see in the questions, pick a reasonable and interesting option that makes sense for this exam, and explain it precisely. Please try to be clear, concise, and reasonably complete, but without wasting time on details that we know you know. We do realize that you may not be able to solve all the problems in the limited time given, but please make sure that what you say be correct and convincing and clearly state what you do not know or are only advancing as a conjecture. We prefer that you formulate your solution in terms of points, vectors, dot and cross products, rather than coordinates.

General (answer 4 of 6)

Stroke Morph

Consider two planar shapes, one in the form of an 's' and the other in the form of an 'e'. Each has been drawn as a variable width stroke with a smooth boundary. Explain how you would formulate the best possible morph between them. We want an animation where the deforming stroke **stays smooth** and **never overlaps** itself. Explain how you propose to measure goodness. Provide a high-level outline of your approach and discuss only the key or tricky detail.
(Hint: Establishing arc-length parameterization of each shape starting from one end is not an optimal strategy. Explain why. If it helps, you may assume that the boundary of each shape is a polygonal loop, but this assumption could make things more complicated for you. If your solution only works under some mild assumption of simplicity for the two shapes, that is fine, but please state them clearly.)

Pentagon Colors

Linear interpolation of colors across a triangle can be done in using barycentric interpolation. Unfortunately, this interpolation approach does not easily generalize to arbitrary polygons, such as pentagons (polygons with five vertices). Propose a method of interpolating colors across a pentagon that matches the color at the vertices and that changes smoothly across the polygon. The results should also be invariant under rotation (the colors do not change if you alter the orientation of the pentagon). Your solution should work for any pentagon, including those that have extreme concavities or that self-intersect.

Universal eclipse

Consider three disjoint balls with centers A, B, C and with respective radii a, b, c. Think of them as the moon, earth, and sun, respectively. Explain how you would implement an algorithm that checks whether there is a total lunar eclipse: no point of the moon A sees any point of the sun C, because the earth, B, blocks it. Please provide the details for the test and the construction/computations involved. We would strongly prefer an exact formulation rather than an approximate answer based on sampling.

Similarity

Suppose a character's pose is represented by a root position in SE(3), two ball joints in SO(3), and 5 revolute joints. How do you measure the similarity of two trajectories of poses? Assume that you know the dimensions of each body part of the character and assume that the length of the trajectory can be different. Come up with two metrics, one of them that is not in Cartesian space, and compare their pros and cons.

Texture

In virtual environments, polygons are typically covered with texture. Commonly, a small example texture is repeated an arbitrary number of times to fill a surface. Consider a simple surface (a plane) and a simple texture (a patch of grass). Even if the texture image is seamless with itself, when repeated many times it will show unnatural cycles that are uncharacteristic of the source texture, e.g.

Even for textures that should be periodic (e.g. a brick wall), if a particular brick texture sample is repeated many times it will be jarring to see the brick pattern repeating (e.g. a dark brick always being below a cracked brick).

We could use various texture synthesis algorithms which can synthesize arbitrarily large textures from small samples without these artifacts, but those enlarged textures might consume too much memory to fit on a GPU.

In the case of our brick wall, we could have a lightweight synthesis algorithm which takes *two* texture patches, each one a single brick, and lays them out stochastically at runtime. Could such an approach work for general textures? Outline an algorithm to build the unit “bricks” or “tiles” from an input texture. Keep in mind that we don’t want visible seams in the resulting assembly of tiles, so adjacent tiles should match at their common border. In addition, we do not want perceptual, low frequency cycles. Also, should there be more than two tiles?

Motion

Provide a brief high-level outline and the math for the formulation of a smooth motion, P(t), of a point in 3D that satisfies the following position constraints P(a)=A, P(b)=B, and P(c)=C and the following velocity constraint P’(b) = B’, where $a<b<c$ and P’ is the derivative of P and where B’ is a vector. In other words, at time t=a, the moving point P is at point A. When time t=b, P is at B and has velocity B’. And when t=c, P is at C. In addition to the math or algorithm for evaluating P(t), explain what kind of a formulation it is and argue why you think this is the best one.

Modeling (answer 2 of 4)

Shortest cycle

You are given a manifold triangle mesh M in 3D that has two bounding loops, A and B, where the vertices are arranged along a regular circle of radius r. M is represented using the Corner Table or some equivalent data structure, so that you can use corner operators: c.v, c.t, c.n, c.s. M is an irregular triangulation of a long, winding tube. Explain how you would compute the key constriction: place where the area of the cross-section of the tube passes by a global minimum. The problem is that the term 'constriction' is not well defined. Please do your best to define it.

Steady edge pattern

You are given two edges $E(A_0,B_0)$ and $E(A_n,B_n)$ in 3D. Explain how you would compute a **steady** pattern of n-1 intermediate edges that starts with edge $E(A_0,B_0)$ and finishes with edge $E(A_n,B_n)$. (A pattern is steady if there exists a transformation T such that $E(A_{i+1},B_{i+1}) = T(E(A_i,B_i))$. Explain your overall approach and provide some guidance (not necessarily all the details) about how to compute it. Is the motion we define unique?

Rooms

M is a triangle mesh with T triangles, E edges, and V vertices. M is connected. Each edge bounds at least 2 triangles. No vertex is a junction (i.e., removing any single vertex will not change the genus or number of components). M (i.e., the closure of the union of its triangles) decomposes three-space into a set of rooms (connected components of the complement of M), including one external, infinite room. Provide an outline and enough details to guide an implementation of an algorithm that computes the number n of rooms. Explain what data structure and operators you would like to have to support your algorithm.

Bent caplet

Consider three disjoint balls with centers A, B, C and with respective radii a, b, c. Explain how they may be used to define a bent caplet: A solid bounded by a smooth surface that contains a portion of Sph(A,a) and Sph(C,c) and is hugging (tangential contact) Sph(B,b) along the way. Provide first a high-level outline and then enough detail to guide an implementation of an algorithm that would render the bent caplet. Explain why the solution of using Neville's algorithm in 4 space (3 coordinates and radius) may fail to produce a smooth surface and how your solution fixes the problem.

Animation (answer 2 of 4)

Drone

A camera-mounted drone (Figure 1) can be modelled as an articulated rigid body system with two body nodes: the “drone node” which can translate and rotate in the world space with 6 global degrees of freedom and the “camera node” which connects to the “drone node” by a 3D $\mathbf{R}^9$ with an upper bound $\mathbf{u}_U$ and lower bound $\mathbf{u}_L$.

You are given a task to take photos of an event in a few spots ($\mathbf{x}_1$, …, $\mathbf{x}_n$) at corresponding time instances (t_1, …, t_n), can you design a trajectory that uses the least amount of control force for the drone to accomplish the task?

Assuming you are able to compute the optimal trajectory, how do you come up with a control “policy” that attempts to visit the same spots at the same time instances?


(Figure 1)

Policy

Tan el al 2014 showed that using Reinforcement Learning method with carefully crafted state, action, and reward function they could tackle challenging motor skill learning problem such as bicycle stunts. In their work, they explicitly staged the learning process from simple tasks to more challenging skills (e.g. learning basic maneuvers first before stunts; separating lateral balance from longitude). Without using these arbitrarily designed learning processes, can you revisit the bicycle stunts problem using the concepts of transfer learning in the recent RL literature? Though this problem is open-ended, please be as concrete as possible when describing the formulation of your RL problem and references of the existing methods you would apply.

(Figure 2)

Closed-Loop Dynamics

Consider an articulated rigid body system with 8 links described in the generalized coordinates (Figure 3).

a) If we define a constraint to connect the last and the first links, how do we compute the constraint force in generalized coordinates (Figure 4)?

b) When the system is in contact with the floor, how do we solve the constraint force in addition to the contact forces (Figure 5)?

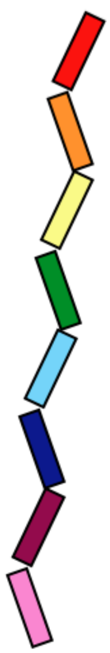


Figure 3

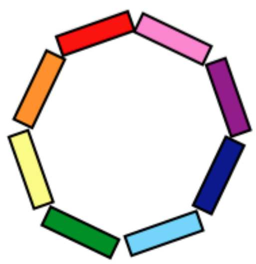


Figure 4

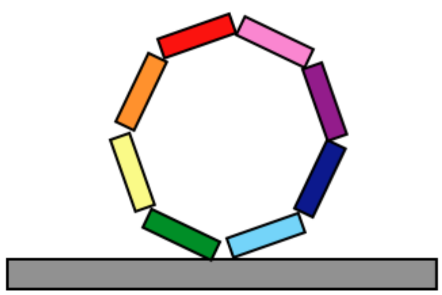


Figure 5

The Elevator

Considering the following scenario. A 50kg human figure enters an elevator (with height = 2.5m) which cable is about to break. As the simulation starts, the elevator and the human figure start to free fall from x meter high. Now, the human figure is well prepared for situations like this because he always wears a jet pack when he rides an elevator. When used, this jet pack provides 500 Newtons of upward force for 0.5 second and is assumed to have no impact on the elevator (the real jet pack needs to follow Newton 3rd Law but let’s simplify things here). However, this jet pack can only be used once so you have to figure out the best time to use it such that you can reduce most of the landing impact and not bump your head at the ceiling of the elevator.

Rendering (answer 2 of 4)

Splashing Water

You are working at a special effects house, and your task is to create a realistic looking water splash. Your boss wants you to test two versions of the rendering software: one using ray tracing and another that uses polygon rasterization. Your renderer should attempt to capture each of the following phenomena: reflection, motion blur, and caustics.
a) How would you achieve these various phenomena using ray tracing?
b) How would you achieve these phenomena using a rasterization renderer?
Do not answer "cannot be done" for any of the phenomena – do your best to come up with approximations if necessary that would fit into the appropriate renderer. No fair using ray tracing in the rasterization renderer!

Anti-Aliasing of Textures

When people talk about texture synthesis, they might be referring to two quite different approaches: 1) solid texture synthesis using composition of functions such as Perlin noise, or 2) texture synthesis from a sample texture image, such as Wei-Levoy pixel-at-a-time or Efros-Freeman patch-based synthesis methods. Depending on which method is being used, quite different approaches to anti-aliasing may be necessary. Describe methods for anti-aliasing each of these two kinds of synthesized textures in turn. Be sure to include in your answer the kinds of filters you would use and why. Also discuss both texture magnification and decimation (viewing from very near and very far).

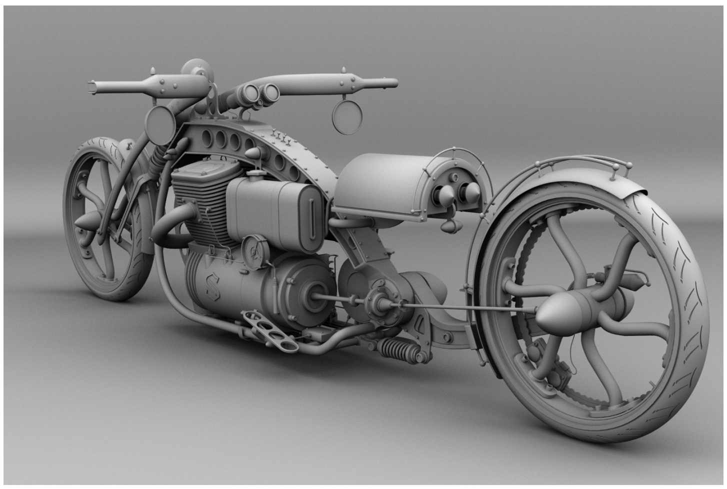
Anatomy Atlas

You are helping a friend create a book of human anatomy in which the illustrations have been rendered directly from volumetric data from real humans. Your "look" will be cartoon-style rendering. Select a direct volume rendering technique (*not* isosurface extraction) and describe how you can write a renderer to provide a cartoon-like style. Your renderer should be able to create the following effects: a) dark silhouettes of objects, b) "flat" cartoon-style shading, and c) hatching that follows the curvature of the surface.

Cloudy Day

A common technique when rendering objects that are outdoors is to include an "ambient occlusion" term. The assumption is that light is reaching a given point from all different directions from the sky hemisphere. This effect is especially important on a cloudy day, when the sun is hidden and a given point is primarily illuminated from light that is transmitted through the clouds. Ambient occlusion creates soft shadows below an object, and creates darker regions in concavities (See Figure 6 below). Answer the following questions about ambient occlusion:

a) Write an equation that describes the reflected light at a point due to ambient occlusion.
b) Describe a naïve ray tracing technique for calculating the amount of light due to ambient occlusion. Don't forget to incorporate the BRDF of the surface that is being illuminated.
c) Describe how you could speed up your method from part (b).


(Figure 6)