Final Exam Review

Computer Graphics
CMU 15-462/15-662, Spring 2016

Notes

- The Final Exam is Monday, May 2nd, 5:30pm-8:30pm in WeH 7500. You will have the full three hours to complete the exam if needed.
- The Final Exam will cover material seen in the course through the entire semester, with some emphasis on material from the second half of the course.
- No notes, calculators, etc. You should not need them. If you find that you do need a formula, etc., ask me and I'll write it on the board.
- These slides summarize the "What you should know" slide at the end of every lecture.
- You can find more practice questions, along with solutions, at the Fall version of the course web site. Note that our lectures did not exactly track those of the Fall semester, so these quizzes may include topics we have not covered and we may have covered topics not represented there. However, it is a great practice resource! http://15462.courses.cs.cmu.edu/fall2015/exercises

Things you should know

Lecture 1: Course Intro.

- For any given setup where we place a camera in the environment, pointing down any of the main coordinate axes (x, y, or z), computing a projection of points in the world onto an image plane.
- Write an algorithm for drawing lines that handles all edge cases (i.e., including edges that are exactly horizontal or vertical).

Lecture 2: Drawing a Triangle / Sampling

1. How should we choose the correct color for a pixel? There is not an exact right answer. However, you should be able to discuss some of the issues involved.
2. What is aliasing, and what artifacts does it produce in our images and our animations?
3. One form of aliasing is where high frequencies masquerade as low frequencies. Give an example of this phenomenon.
4. Suppose we have a single red triangle displayed against a blue background. Does this scene contain high frequencies?
5. What does the Nyqvist-Shannon theorem tell us about how image frequencies relate to required sampling rate?
6. The practical solution on your graphics card for reducing aliasing (i.e., for antialiasing) is to take multiple samples per pixel and average to get pixel color. Try to use what we learned about sampling theory to explain as precisely as you can why taking multiple samples per pixel can reduce aliasing artifacts.
7. Be able to write an implicit representation of an edge given two points.
8. Be able to use the implicit edge representation to determine if a point is inside a triangle.

Lecture 3: Math for Graphics and Transforms

1. Express points and vectors using homogeneous coordinates.
2. Perform a dot product and demonstrate how to use the dot product for projection (e.g., projection of a point onto a coordinate axis).
3. Perform a cross product and demonstrate how to use it to compute a surface normal.
4. Perform matrix multiplication.
5. Derive a parametric expression for a line between two points.
6. Prove that your parametric expression is correct. Discuss whether it is unique.
7. Derive an implicit expression for an edge between two points.
8. Prove that your implicit expression is correct. Discuss whether it is unique.
9. Use an implicit edge expression to determine whether a (2D) point is inside a (projected) triangle.
10. Create an algorithm to rasterize a triangle from a set of "inside-triangle" tests.
11. Make that algorithm efficient so that not every pixel needs to be tested.

Lecture 4: More Transforms

- Create 2D and 3D transformation matrices to perform specific scale, shear, rotation, reflection, and translation operations
- Compose transformations to achieve compound effects
- Rotate an object about a fixed point
- Rotate an object about a given axis
- Create an orthonormal basis given a single vector
- Understand the equivalence of [x y 1] and [wx wy w] vectors
- Explain/illustrate how translations in 2D (x, y) are a shear operation in the homogeneous coordinate space (x, y, w)

Lecture 5: Projections

- Form an orthonormal basis
- Create a rotation matrix to rotate any coordinate frame to xyz
- Create the rotation matrix to rotate the xyz coordinate frame to any other frame
- Rotate about axis w by amount theta
- Know basic facts about rotation matrices / how to recognize a rotation matrix
    - Rows (also columns) are unit vectors
    - Rows (also columns) are orthogonal to one another
    - If our rows (or columns) are u, v, and w, then uXv=w
    - The inverse of a rotation matrix is its transpose
- Create a projection matrix that projects all points onto an image plane at z=1
- Propose a projection matrix that maintains some depth information
- Understand the motivation behind the projection matrix that projects the view frustum to a unit cube
- Be able to draw / discuss the details of the view frustum

Lecture 6: Barycentric Coordinates

- Interpolate colors using barycentric coordinates
- Compute barycentric coordinates of a point using implicit edge functions
- Compute barycentric coordinates of a point using triangle areas
- Estimate the location of a point inside a triangle given its barycentric coordinates
- Estimate the location of a point outside a triangle given its barycentric coordinates
- Estimate barycentric coordinates of a point from a drawing.
- Show that interpolation in 3D space followed by projection can give a different result from projection followed by interpolation in screen space. In other words, explain why interpolation using barycentric coordinates in screen space may give a result that is incorrect.
- How, then, can we obtain a correct result using interpolation in screen space?

Lecture 7: Texture

- Textures are used for many things, beyond pasting images onto object surfaces.
  - Normal maps (create appearance of bumpy object on smooth surface by giving false normal to the lighting equations)
  - Displacement maps (encode offsets in the geometry of a surface, which is difficult to handle in a standard graphics pipeline)
  - Environment maps (store light information in all directions in a scene)
  - Ambient occlusion map (store exposure of geometry to ambient light for better representation of surface appearance with simple lighting models)
  - Can you think of / discover others?
- Know how to interpolate texture coordinates
- Know how to index into a texture and compute a correct color using bilinear interpolation
- Be able to create a mipmap and store it in memory
- Be able to compute color from multiple levels of mipmaps using trilinear interpolation
- What is the logic behind selecting an appropriate level in a mipmap?
- What can happen if we select a level that is too high resolution? too low resolution?

Lecture 8: Graphics Pipeline

- What is the depth buffer (Z-buffer) and how is it used for hidden surface removal?
- Where does the depth for each sample / fragment come from? Where is it computed in the graphics pipeline?
- Is the depth represented in the depth buffer the actual distance from the camera? If not, what is it?
- What is the meaning of the alpha parameter in the [R G B a] color representation?
- Be able to use alpha to do compositing with the "Over" operator.
- Is "Over" commutative? If not, create a counterexample.

$$C = \alpha_B B + (1 - \alpha_B)\alpha_A A$$

- What is premultiplied alpha, and how does it work?
- Be able to use premultiplied alpha for "Over" composition.
- Why is premultiplied alpha better?
- How do we properly render a scene with mixed opaque and semi-transparent triangles? What is the rendering order we should use? When is the depth buffer updated?
- Draw a rough sketch of the graphics pipeline. Think about transforming triangles into camera space, doing perspective projection, clipping, transforming to screen coordinates, computing colors for samples, computing colors for pixels, the depth test, updating color and depth buffers.

Lecture 9: Introduction to Geometry

- List some types of implicit surface representations
- What types of operations are easy with implicit surface representations?
- List some types of explicit surface representations
- What types of operations are easy with explicit surface representations?
- What is CSG (constructive solid geometry)? Give some examples of CSG operations.
- What type of representation is best for CSG operations?
- Describe how to do union, intersection, and subtraction of geometry using simple operators on a surface representation.
- What is a level set representation? When is it useful?
- What types of splines are common in computer graphics?
- Why are they popular? What properties make them most useful?
- Be able to use an expression such as $u^3p_0 + 3u^2(1\text{-}u)p_1 + 3u(1\text{-}u)^2p_2 + (1\text{-}u)^3p_3$ or its matrix equivalent to understand properties of a cubic spline or sequence of cubic splines. Think about continuity at the join point between two splines, whether it interpolates its control points, whether it has the convex hull property.
- What was one motivation behind developing the rational b-spline? (representing conics)
- What is one advantage of using a *non-uniform* rational b-spline? (control point “strength”)

Lecture 10: Curves, Surfaces and Meshes

- How to use split and average operations to do subdivision
- What is a manifold surface?
- Distinguish manifold from non-manifold surfaces
- Can a manifold surface have a boundary? Give an example.
- Explain the idea of surface curvature with a diagram.
- Give an example of a surface where one of the principal curvatures is zero
- What do you need to store in a halfedge data structure?
- How can you find all vertices in a face with this data structure?
- How can you find all faces that contain a vertex with this data structure?
- Be able to perform edge flips, edge splits, and edge collapse with this data structure.
- BONUS: Think of an algorithm to traverse every face in a manifold using this data structure.

Lecture 11: Geometry Processing

- List practical applications that you can relate to for good geometry processing algorithms.
- List some rules of thumb for good quality meshes
- How can we test if a mesh is manifold?
- Give examples where edge flip and edge collapse may make a manifold mesh into one that is no longer manifold
- Give pseudocode for one iteration of Loop subdivision
- Explain the important properties of various subdivision algorithms (interpolation, continuity, behavior at the boundaries)
- Be prepared to calculate vertex updates in a simple example of Loop or other subdivision. (The vertex weighting masks will be given to you.)

Lecture 12: More Geometry Processing

- Express distance from a plane, given a point on the plane and a normal vector
- Show how the Q matrix (the quadric error matrix) represents squared distance from a plane
- If we have a Q matrix and a point, how do we calculate cost?
- How does this cost represent distance to the original surface?
- Given Q matrices encoding triangles in a mesh, how do we get the Q matrix for each vertex?
- If we collapse an edge, what is the Q matrix for the new vertex that is added in the edge collapse?
- Describe some techniques for improving the quality of a mesh to make it more uniform and regular.

Lecture 13 was the Midterm Review

Lecture 14: Radiometry

- Visible light consists of a small range of wavelengths along the spectrum from gamma rays to radio waves.
- Energy of a photon depends on wavelength (and speed of light, and Planck's constant)
- Roughly, what is the spectral power distribution of different familiar light sources?
- What is flux?
- What is irradiance?
- Why (how) does irradiance depend on the angle between the light source and a patch of surface area?
- How does irradiance fall off with distance?
- What is a solid angle?
- What is radiance, and how many dimensions do you need to capture the radiance in a scene (i.e., to capture a light field)?
- What is radiant intensity?
- You should know how to read a Goniometric diagram.
- What is the "circle of confusion" and what visible effect does it produce in a photo?

Lecture 15: Integration

- How do we use the trapezoidal rule to integrate a function?
- How does work increase with dimensionality of our function?
  - This is why we typically use Monte Carlo integration in graphics!
- Give a high level overview of the process of Monte Carlo integration
- What is a probability density function (PDF)?
- What is a Cumulative Distribution Function (CDF)?
- The Inversion Method can be used to correctly draw a sample from a PDF.
  - Sketch the overall step by step process for using the Inversion Method.
  - Work through how to use it to sample area of a circle
  - Work through how to use it to sample solid angles from a hemisphere
- What is rejection sampling? Show how to use rejection sampling to sample area of a circle, volume of a sphere, directions on a sphere, and solid angles from a hemisphere.
- Use one of the sampling methods we discussed to correctly accumulate incident irradiance on a surface patch.

Lecture 16: Materials

- Draw diagrams to illustrate reflection for (1) a perfectly specular (mirror) surface, (2) a glossy surface, (3) a diffuse surface, (4) a retroreflective surface
- What is a BRDF? What are the inputs and outputs of a BRDF function?
- How would you measure a BRDF? How many degrees of freedom would you need?
- Given a ray and a surface normal, calculate the direction of perfect reflection
- Given a ray, a surface normal, and indices of refraction, calculate the direction of perfect transmission using Snell's Law.
- What is Fresnel reflection. Sketch curves to illustrate the effect for dielectrics vs. conductors as we have seen in class. Label your axes. Informally, what does this effect show?
- What is subsurface scattering?
- How can we extend the idea of BRDF to subsurface scattering? What additional parameters must be sampled?

Lecture 17: Accelerating Geometric Queries

- Compute ray-triangle intersection, including checking whether the ray passed through the inside of the triangle.
- Compute ray - bounding box intersection
- Construct a bounding box hierarchy for a given collection of objects.
- Calculate traversal order of a bounding box hierarchy for a given ray.
- What is the Surface Area Heuristic (SAH) and what goals is it trying to achieve?
- Explain how to choose a bounding box partition using the SAH
- Be able to distinguish between object-centric (primitive partitioning) acceleration structures and space-centric (space-partitioning) acceleration structures
- Know the difference between these acceleration structures, how to build them, how to traverse them, and when to use each type:
  - bounding box and bounding sphere hierarchies
  - KD-trees
  - octrees
  - grids

Lecture 18: Global Illumination

- Explain and illustrate with a sketch the emitted light term, the direct light term, and the indirect light term in the Rendering Equation.
- Be prepared to define, explain, and illustrate all terms in the Rendering Equation if it is presented to you in any of the various forms we have seen in this lecture.
- When we are taking samples of multiple variables, how can the idea of "Splitting" help to save work? Give an example from path tracing.
- What is Russian Roulette? How does Russian Roulette help us to obtain unbiased samples in path tracing?
- If we choose to continue tracing a ray in a Russian Roulette scenario, we will reweight the value provided by this ray by some factor. What is the weighting factor and why is it used?
- Form factors fall out of rewriting the Rendering Equation from a perspective focusing on small portions of area. What exactly does the Form Factor represent? Copy out the form factor expression (G in slide 43), draw an illustration, and explain why each of the various terms must be included.
- Another point of view on the Rendering Equation is to sum up light energy contributions from emitted light, paths of length 1, paths of length 2, etc. Be prepared to explain what effects you would see when tracing out paths of various lengths (color bleeding, reflections, caustics, etc)

Lecture 19 (1 of 2): Advanced Rendering

- **Be familiar with the following expression for Monte Carlo integration. What is the role of each term?**

$$I = \lim_{n\to\infty} V(\Omega)\frac{1}{n}\sum_{i=1}^{n} f(X_i)$$

- **What is expected value?**
- **What is variance?**
- **Give an example of how we can reduce variance in our rendered results in a path tracing algorithm without increasing the number of samples.**
- **What does it mean for an estimator to be consistent?**
- **What does it mean for an estimator to be unbiased?**
- **Give a concrete example of how a renderer could give a biased estimate of an image. Is the renderer in your example consistent? Explain your answer.**
- **Give five examples of how you can reweight samples in a pathtracing algorithm in order to do importance sampling.**
- **What are the main ideas behind bidirectional path tracing?**
- **How would you enumerate all possible paths in a scene?**

Lecture 19 (2 of 2): Advanced Rendering

- How does Metropolis-Hastings sampling work?
- Assume you have code to generate random paths and code to mutate existing paths. Write pseudocode for Metropolis-Hastings path tracing.
- Is this algorithm consistent? Is it unbiased? Is it efficient? For what kinds of scenes would this algorithm be best suited? Explain your reasoning for your answers to all of these questions.
- What is Stratified Sampling?
- Why is it preferred to random sampling?
- Hammersley and Halton points are pseudo random sampling techniques to generate points with low discrepancy. What is discrepancy? Why do we want to generate low discrepancy samples?

Lecture 20: Fast Ray Tracing

- Be able to describe the main ideas and some strengths and weaknesses of photon mapping and radiosity.
- Why does radiosity give a biased estimator? an inconsistent estimator?
- How does Shadow Mapping make use of rasterization in the shadow computation?
- How do we use rasterization to capture reflections in an environment map such as a cube map?
- Give an example of where an environment map gives a poor approximation.
- What is an ambient occlusion map? How does it fit into the graphics pipeline (i.e., give a rendering flow that makes use of an ambient occlusion map to speed computation).
- Give some examples of scenes or situations where these tricks are insufficient, and we can achieve a benefit from instead having a real time ray tracer.
- Understand the "rules of the game" for high speed ray tracing: multiple cores, SIMD processing, and high cost of memory access.
- What does it mean to trace a ray packet? Under what circumstances will ray packet tracing be efficient? ..be inefficient?
- What sorts of things would you think about if writing a scheduler to determine the order in which rays will be traced?

Lecture 21: Animation

- Be ready to talk about history of animation. Say briefly how and when some important advances in animation occurred.
- Why do we perceive sequences of still frames as motion?
- What is in-betweening?
- Keyframing, motion capture, and procedural animation are three animation techniques. Give examples of each (e.g., from movies, research, or possible class projects). Give some pros and cons of each.
- Define interpolation, continuity considerations, and locality for splines. Can cubic splines give us interpolation, C2 continuity and locality all at once?
- If we give up locality, what type of spline can we use? What if we give up C2 continuity? If we give up interpolation?
- What are some things you found interesting about the professional character rig that add to its functionality and complexity?
- What are blend shapes and how are they used?

Lecture 22: Dynamics and Time Integration

- Sketch a block diagram of a physically based simulation engine. Include modules for integrating the system forward in time, for calculating derivatives, for obtaining forces.

- How did you express state?

- Show state, and exact functionality of each module for a system where two 2D particles are connected by a spring. You may use forward Euler integration.

- The Euler-Lagrange equation is listed below. Explain each of the terms. Use the Euler-Lagrange equation to find the equations of motion for a particle acting under gravity, two particles connected by a spring, other simple systems.

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}} = \frac{\partial \mathcal{L}}{\partial q}$$

- What equations are used for forward Euler integration? Forward Euler integration can suffer from two kinds of problems: (1) poor accuracy, and (2) instability. Explain these two problems. Illustrate each problem with a sketch.

- How do you set up Backward Euler integration? Is Backward Euler stable? Is it accurate? What problems may be observed when using Backward Euler? (Alternatively, what are the pros and cons of using Backward Euler?)

- Be prepared to explore other integration schemes, perhaps integrating a simple system forward a couple of steps by hand.

- We would typically use symbolic differentiation in a simulator, but sometimes it is convenient to use numerical differentiation. How would you numerically differentiate a given function f(t)?

Lecture 23 (1 of 2): Color

- **Why is it that we can get away with just three values for color (e.g., R-G-B)?**
- **What are the rods and cones, and how many types of each are there in the human visual system? What can you say about how they are distributed in the retina?**
- **We discussed a color matching experiment which had the goal of matching a given wavelength of color by tuning three different lasers. What is the goal of this experiment? What can we learn from it?**
- **If we plot the results of this experiment by wavelength, we obtain the trichromatic matching functions. Roughly sketch what these look like for the RGB color space we have been using in this class.**
- **An alternative to RGB color space is the CIE color space, with X, Y, and Z primaries. Roughly sketch the X, Y, and Z primaries by wavelength. What is Y in this color space? What problem with RGB color space does the CIE color space solve?**
- **Given a color space expressed by some three-dimensional basis, can it be converted into any other basis through a linear operation? (True or False)**

Lecture 23 (2 of 2): Color

- Perceptual color spaces such as Y'CbCr express luminance in a way such that equal steps in luminance appear as equal steps in brightness to our visual system. How does perceived luminance Y' relate to actual luminance that can be measured as light source energy?
- What is gamma correction? Give three examples where gamma correction is useful.
- What is high dynamic range imaging?
- How could you capture high dynamic range information about a scene in the real world?
- What is tone mapping?
- Can you think of a good algorithm for tone mapping? What would you want to preserve?

Lecture 24: Image Processing

- What is the flow of operations involved in JPEG compression? How does JPEG compression achieve reduced storage space? What kinds of artifacts can be expected to result?
- Show examples of 3x3 blur, sharpening, and edge detection filters. Be able to generalize these ideas (e.g., create a filter to detect diagonal edges).
- Why is a Gaussian filter preferred to the box filter for creating blur? (You may want to refer back to the beginning of the course.)
- How does the median filter work? What is it designed to achieve?
- How does the bilateral filter work? What is it designed to achieve?
- We discussed a technique to de-noise images using information from other parts of the image (specifically, pixels having similar local neighborhoods). Explain this approach.
- We also discussed a non-parameteric texture synthesis technique that similarly makes use of neighborhood information to fill in empty pixels. Give pseudocode for such a technique.
- Which of the following filters use convolution? If a filter does not work through convolution, explain why not. The filter types are: blur, median, sharpen, edge detection, bilateral.

Lecture 25: Lightfield

- Give examples of the types of computation that may be done within the camera itself to assemble and improve the final generated image.
- What are sources of noise in capturing / measuring images in a typical digital camera?
- How is color handled in a typical digital camera (e.g., explain the Bayer mosaic)?
- What does it mean to “demosaic” an image?
- Give some examples of how an image may be improved using outputs from multiple cameras (e.g., as in the Light L16 camera).
- Give a 4D lightfield parameterization of rays entering a camera using coordinates on the image plane and on the camera lens.
- How does the microlens array of a lightfield camera allow this 4D space to be captured within a single 2D array? Draw sketches to illustrate your answer.
- What are the advantages to capturing a 4D representation of the light energy in a scene vs. a standard 2D image representation?
- What are the disadvantages?