

CS 184: Assignment 6 — Final Project

Ravi Ramamoorthi

Goals and Logistics

The final project is a chance to showcase what you've learned in the course, to develop a project of your choice. This can be done in groups of two or individually as you prefer (if you have a partner you work well with, we recommend you use groups of two to produce a more interesting project, but if you want to go it alone or don't have a suitable partner, that's fine as well). We fully understand that you have only about 3 weeks for this, that also needs to be used in studying for finals. As such, we will have reasonable expectations of what can be achieved. Note that this assignment will be graded on a more *subjective* curve. Something that satisfies the "basic" requirements below will obtain about 80% of the score. Higher scores will be reserved for the best assignments; extra credit will also be possible. The class website includes links to the best projects for last semester and a couple of years back, which you can use as a guide.

The main goal of this assignment is that you should extend one of the homework assignments (of your choice) to add new functionality, and make the assignment really nice. (This would roughly be the extra credit part; if you already got extra credit though, you need to do more or pick a different assignment). We also give ideas for an animation project, and are open to other possibilities (that is, you can propose any project of your choice; simply ask the instructor and teaching assistants).

I again want to emphasize that since time is short and you have numerous other issues to deal with, don't go overboard unless you really have the time. A very simple project that addresses some of the extra credit portions of the previous assignment is more than adequate. I view this as an easy way to make one really nice assignment in the course, and get some points, rather than as a terrible last-minute struggle. You should certainly not aim on spending as much time on this as in homeworks 4 or 5. Obviously, given the timeframe, we expect that you are familiar with the base assignments, and therefore there is little support (no skeleton code). You may of course reuse any material you wrote from earlier in the course.

Since this is a relatively open assignment, your submission ("submit as6") should just include a *README* that links to a website and has any other pressing information of interest (also make sure it includes the name of your partner or if you worked alone). Both partners should submit (the *README* file and of course the website can be identical). Submission of source/executable code is optional, but your assignment does need to be gradable. (If there is not adequate visual documentation, nor code, there is no easy way for the teaching assistants to see what you've done). Do not modify the website after the due date.

Since this assignment is more open, grading will be based on a qualitative sense of what you accomplished. Therefore, the website and writeup is really crucial, as we will not have time for demo sessions. Hence, make sure it fully documents what you did, and includes example images (video should be included if you are showcasing a real-time system or animation of any kind; it may be omitted only when your only output is still images). In the future, whether you go to graduate school or write reports in industry, your work will be evaluated in many instances by the written documentation; thus it is a good exercise to make sure your website adequately documents your work to the extent we can grade it. Clarity and completeness of documentation will be factors in assigning the final score; in particular there may be significant deductions if the GSI cannot understand or evaluate what you did from your website.

Your submission must be on time, since grades need to be submitted soon. As such, a stronger late policy applies than the default for the course, and the instructor reserves the right to not count (or very heavily penalize) late submissions. Plan ahead and turn in what you have.

Project Ideas

Here are some project ideas (note that you only need to do one, only need to add one or two features, and we strongly recommend you build on one of the earlier assignments, effectively doing the extra credit

portion of that). These ideas are much less specified than in earlier homeworks. This is because the project is open-ended, and you should feel free to exercise your creativity (just make sure to document what you actually did when you turn in your submission). You are also free to propose a comparable separate or related project of your interest; does not have to be the suggestions on the next page. At the same time, please bear in mind again that you need/should not go overboard, and doing one or two extra credit features of an existing assignment is adequate. Balance your time with preparation for finals wisely.

- *Scene Viewer:* Make your scene viewer program from assignment 2 really fancy, including some of the extra credit options (or more if you already did them). In particular, include support for more complete scene graphs and instancing and hierarchy. Include support for more geometry, at least the triangle meshes with normals supported in homework 5. Include a fancier lighting and shading model, such as spot lights, environment maps and so on. Include support for picking and manipulating individual objects, let the user zoom into the scene and translate, rotate or scale objects. Make something of the usability of *iview*, an SGI software from a time long ago (look online for examples of use). You do not need to do all of the above suggestions; a couple of well chosen items should suffice.
- *Curves:* If you have not already done the extra credit, an obvious extension of homework 3 is to the extra credit part, to take a curve and generate a surface of revolution. An alternative project is to extend the B-Spline curve editor to go from uniform to non-uniform, possibly rational NURBS. There are many other possibilities, such as drawing a variety of surfaces beyond just those of revolution. The idea is to make a more complete modeling package (note that only one of these extensions is required; you do not need to do all).
- *Interactive Rendering:* The OpenGL assignment in homework 4 can easily be extended with many of the extra credit items listed. For the really ambitious students (this is hard; you should try only if you've done exceptionally well in the class or happen to have a lot of motivation and free time), you could try extending the homework into a video game with interactive gameplay. Given the timeframe, it would probably need to be a relatively simple game.
- *Raytracing:* There are numerous extension possibilities for raytracing, such as those listed in extra credit. Try to do at least one of optimized acceleration structures (if you didn't already), more primitives (cubes, cones, spline surfaces would be nice but not essential), distributed ray tracing for area lights and soft shadows, motion blur or depth of field. The really ambitious among you might also try to write a global illumination renderer, at least for diffuse surfaces, by bouncing photons.
- *Animation:* Unlike the previous topics, there is no base animation assignment to extend, and you need to start from scratch. As such, I would recommend this only if you feel like you have a lot of time available, have already done exceptionally well in the course, you work in groups of two, and you really have strong interest in animation.

The goal is to build a basic inverse kinematics system. You may want to start with the analytic formulae for a two-joint arm, but you would ideally like to have a general solver for at least four joints (and preferably with different lengths). The root should be stationary (attached to an immobile base). There should be some interactive or offline mechanism to specify the goal point in 3D (or create a procedural animation), and the IK solver should select joint angles to reach the desired goal. For submission, make a video that clearly shows your system working (and any problems if relevant), where the goal path is something complicated, that tests the full range of motion (a circle or large arc or figure-eight, not just a line). It would also be nice if you gracefully handled configurations that cannot be achieved (are out of reach). Make sure to try to cover the full range of the arm's operating space in terms of joint angles. Clearly, a full video documenting everything that was done is crucial. Note that the rendering can be very simple, reusing homework 4 (or 5) or any other relevant code; we want to focus on the animation.