Teaching Statement for Matthew Kaplan

My primary reason for seeking an academic position is that I enjoy teaching and interacting with students. I feel that teaching done with dedication and energy can be a rewarding experience for both professor and student. I firmly believe that a quality education is fundamental for students and society; each benefits from the sharing and advancement of knowledge.

It is critical to challenge and excite students at all levels. I develop lectures and projects that incorporate and make reference to real world, historical and cross-disciplinary applications, using modern computer applications as examples that help motivate course work. I try to give students open-ended projects that cover the relevant theory yet that allow them some individual choice in the content. Some modern pedagogical theory says that students have a lecture attention span of about 20 minutes. Therefore, I like to break lectures up into small sections, punctuated with demonstrations, feedback and discussions with both peers and myself.

One of my favorite professors was fond of saying, “If you can't explain it to a ten year old, you don't understand it.” Effective teaching begins with an in depth understanding of the material and a clear and efficient lesson plan. However, I understand that no single lesson plan is sufficient for all students. It is clear to me that students learn differently and that gauging the class is an important skill. As each course progresses, an individual assessment should be made of every student to try to learn which teaching techniques are being effective for students at every level. I emphasize frequent student/teacher interaction and feedback through in class discussion, one-on-one discussions of individual projects and course work, and provide daily out of class availability for students who may need extra help.

It is important that a student learn *how to learn*, as a general and continuing principle. Students without this ability will be handicapped as they find their skill set becoming rapidly outdated as the field evolves around them. Lifelong learning is a characteristic which is important in *all* facets of our lives. While this may be beyond the scope of any single class, it should be emphasized and evident in *every* class. A teacher is offered a unique position to engage students, yet should not ignore their responsibility to teach students the spirit of personal inquisitiveness that leads to being engaged in the absence of a teacher. I believe that a course taught with an emphasis on problem solving and creative thinking abilities can challenge students and show them that they are capable of tackling a course of any level. In pursuit of this, I try to maintain a good balance between theory and practice by presenting specific material while also attempting to frame the current problem in the larger context of the subject being studied.

Computer science is a field that by its nature evolves rapidly and whose newest hardware is half-speed in a mere eighteen months. Therefore, I believe it is important to constantly stay up-to-date with recent developments in my field and in computer science as a whole. Students have more confidence in a teacher who is well informed and will benefit by receiving information that matches the current state of the technology. Also, the research and community life of a teacher is a valuable addition to the academic life of the students. I attempt to motivate students with examples of research being done at the university and try to get them to collaborate on undergraduate research projects when possible. Students who find their professors to be both personally and academically accessible will be more excited by the field and more prepared to evaluate their graduate school options. .

Teaching Experience

As an undergraduate at Bowdoin, I was asked by one of my professors to teach a series of classes on internet programming for advanced high school students as part of a larger community outreach program the college was doing. This was a series of six lectures each of which was accompanied by a project.

During my time as a graduate student at the University of Utah, I was a research fellow and therefore had no formal teaching experience. My advisor did not use a TA for her CAGD course and I was asked to fill this role in an informal capacity numerous times over a five year period. This included preparing course materials, lesson plans, lecturing, assigning and evaluating projects, and grading as well as teaching the same set of two lessons for four years in a row. I provided informal office hours and allowed students to schedule meeting times with me for help and instruction. I also taught several Introduction to Graphics classes.

As a postdoctorate, I was encouraged to involve the students in my projects and therefore I had the chance to work closely with several undergraduate students and new graduate students. This included formal instruction, lectures and discussion sessions for groups of between 1-5 students. For two students, in particular, I was able to devote a significant amount of time to teaching advanced topics in Non-Photorealistic Rendering and developing research projects with them. .

Teaching Courses

With my computer science experience, I believe I am qualified to teach any core computer science classes though clearly, I would be most qualified to teach topics relating to computer graphics. Here is a list of several example courses I feel most qualified to teach.

Introductory Computer Science Courses – introduction to CS, discrete math, data structures, programming languages, C, C++, Java, computing for engineers, software engineering, foundations, theory of computation, etc.

Introduction to Graphics – pipelines, rasterization, lighting, visibility, viewing, transformations, blending, sampling theory, texture mapping, shadows, ray tracing, API's (OpenGL & DirectX)

Advanced Computer Graphics – advanced ray tracing, visual perception, procedural texturing, image based graphics, speedup issues (BSP trees, etc), environment mapping, soft shadows, vertex and fragment shaders, radiosity, lightfields

Non-Photorealistic Computer Graphics – g-buffers, silhouettes, suggestive contours, ridge/creases, artistic strokes, animating contour motion, halftoning, dithering, screening, stippling, natural media simulation, rendering illustration, image and object space distortions, architectural illustration, technical illustration, medical illustration, paint stroke simulation and planning, multi-scale stroke selection

Computer Graphics Mathematics – linear algebra, discrete mathematics, numerical analysis, signal analysis – convolution, filtering,etc., probability, computational geometry, wavelets

Computer Vision / Image Processing – perception and color theory, sampling, signal processing, noise, cameras, displays, convolution, smoothing, morphological operators, histograms, differential operators, feature extraction, scene reconstruction, image restoration, recognition,

Scientific Visualization – volume rendering and isosurfaces, transfer functions, visual perception

Computer Aided Geometric Design – differenential geometry, conics, curves (implicit, Bezier, Bspline, Hermite), Bezier and Bspline surfaces, subdivision surfaces, NURBS, approximation techniques,

Art and Computers – human computer interaction (HCI), issues in automating vs. manual control, style analysis and control in modeling, rendering and animation, visual perception, pictorial communication, data visualization,