COVER FEATURE

Building a Game Development Program

Lawrence Argent, Bill Depper, Rafael Fajardo, Sarah Gjertson, Scott T. Leutenegger, Mario A. Lopez, and Jeff Rutenbeck
University of Denver

In 2004, the University of Denver began offering majors in game development, with research and scholarship directed toward "humane gaming," a term that includes medical and educational game applications and socially conscious gaming.

At the University of Denver (DU), we believe game development is an interdisciplinary field requiring some appreciation for both the technical and creative, with a deep passion in at least one area. Our two new game development majors are built upon a four-way partnership between computer science, digital media studies, electronic media arts design, and studio art.

We created our program with the following goals in mind:

- Developing a humane emphasis. We have coined the phrase "humane gaming" to describe our efforts to direct this new academic discipline into benevolent applications. As an intended side effect, we conjecture that this emphasis will help to attract women to the computer/videogames development field.
- Fostering "whole brain education." By this we mean developing students' abilities in both analytical and creative thinking. We believe that many students have both types of skills without knowing it. Further, we believe that exercising both abilities results in deeper understanding and intuition than is possible with only one type of thinking.
- Increasing enrollments in computer science and developing a technically strong major.
- Allowing student interest driven balancing between computer science, digital media, electronic art, and studio art.

INTERDISCIPLINARY PARTNERS

Achieving our program's goals requires strong collaboration and an interdisciplinary approach.

Computer science

The DU Computer Science Department administers our new game development major. The department offers BS/BA, MS, and PhD degrees in computer science.

Our undergraduate computer science BS degree follows a model similar to the ACM guidelines. Therefore, the department already offers most of the technical classes needed for the game development degree, including introductory programming, discrete mathematics, algorithms and data structures, computer organization, operating systems, and graphics, as well as elective topics such as multimedia, computational geometry, networks, databases, software engineering, distributed systems, theory of computation, and security.

Computer science forms the fundamental core of our degree program. Our intent is to produce graduates who are ready for the technical side of game development as well as many other careers within computing and information technology. Students who choose our BA and double major in Art/Electronic Media Arts Design/Digital Media Studies might follow career paths that are not specifically on the technical side, but they will leave our program with a deep understanding of the technical side of computer science and game programming.

Published by the IEEE Computer Society

Digital Media Studies

Founded in 1996 as a partnership between the School of Communication, the School of Art, and the Department of Computer Science, the DU Digital Media Studies program forms the basis of content development for the game development program. With its wide range of offerings at the undergraduate and graduate levels, DMS involves students in a design-technical-critical approach to digital media.

The design curriculum within DMS, which is a shared curriculum with the Electronic Media Arts Design (EMAD) program, lets students explore the many aspects of communicating visual messages through courses such as Introduction to EMAD, Net Art and Design, Interactive Art and Design, Identity and Branding, Typography, and Designing Social Awareness. All DMS students are required to complete the first two segments of the art foundation experience, then they pursue additional study in digital design—a minimum of four design courses.

The DMS experience prepares game developers for the complex creative and development culture they are likely to encounter in the workplace.

The technical portion of the DMS curriculum corresponds with five general production areas of digital media: Web/Internet, interactive media/games, digital video production, digital audio production, and 3D modeling and animation. Courses in the video, audio, and modeling/animation areas expose students to a wide range of hardware and software techniques such as video/audio acquisition, editing, and character animation. Advanced courses in the Web/Internet and the interactive media/games areas combine similar hardware and software instruction with extensive work in programming for digital media—for example, Web applications, rich Internet applications, personal media, and 2D game development.

The critical DMS curriculum examines the broader implications of digitization, especially regarding computer-mediated communication, social networking, privacy, digital media ethics, globalization, impact on traditional media, social impacts of digital media, futurism, and so on. All DMS students are required to take a foundation course in critical approaches to digital media studies, then they pursue advanced critical coursework in more specific topics.

In a general sense, the design-technical-critical approach forms a strong complement to the game development degree programs. While the game development curriculum focuses on the technical theory and skills needed to program the intelligence, performance, and interactivity of games, the DMS program incorporates content development skills and sensibilities into the process. As a well-balanced cross-disciplinary program that exposes students to a wide range of professional practices, the DMS experience also prepares game developers for the complex creative and development culture they are likely to encounter in the workplace.

Studio Art and Electronic Media Arts Design

DU's School of Art and Art History is committed to teaching the technical and creative processes of art, methods of analyzing and criticizing visual culture, and the diverse history of art. Students are educated to think critically, engage in the art-making process, express themselves creatively, and articulate their ideas clearly. The School of Art and Art History offers three primary domains of study: Studio Art (SA), Art History, and the EMAD program. All students in the SA and EMAD domains take the Core Art and Media Program (CAMP), a common introductory sequence.

Studio Art. In the SA program, students select from a range of studio courses: ceramics, drawing, painting, photography, printmaking, and sculpture. Students with multiple interests who want a liberal arts education with a major in art and a second major or minor in another field usually complete the program for a BA degree. Students who intend to become professional artists or art teachers or who want a more thorough art training usually complete the program for a bachelor of fine arts (BFA) degree.

The BA in SA combines CAMP, art history, and SA classes, resulting in a total of at least 60 art credits. The BFA in SA combines CAMP, art history, SA, and EMAD, resulting in a total of at least 110 art credits. Admission into the BFA program requires portfolio approval. Due to the intensive art requirements of the BFA, it would be impossible to double major with game development in four years, and we expect few students to pursue this option.

Electronic Media Arts Design. The EMAD program produces self-motivated artists with strengths in creative vision, design principles, the use of computers and design software, concept development, critical thinking, and a commitment to communicating visual messages capable of reshaping the world through innovative artistic design. The program provides a background for students who want to enter the design profession or who are interested in pursuing an independent career as exhibiting electronic media artists.

EMAD courses emphasize the interrelationships between humans and computation. The electronic design curriculum is biased toward the pragmatic concerns of communication, while electronic arts curricula tend toward the more exploratory, addressing "what if" questions. The design component is a way of thinking that employs the iterative methodology of refinement, an important aspect of how games are created.

This content-focused education develops thoughtful game makers who will benefit the industry and society. EMAD is in an interesting position to bring to fruition these lurking skills and abilities.

Flexibility

Our experience with college freshmen is that many incoming students are unsure about their intended major and often change majors within the first year or two. Our program is set up to allow students to experiment and find the path that best fits their interests.

Students typically take four classes per quarter, and incoming freshmen usually choose first-year English and a foreign language as two of those classes, thus leaving two courses per quarter for our major.

Incoming students who indicate they are more interested in the technical side are encouraged to take the introductory programming sequence and calculus, deferring art to the second year. If students decide they want to pursue our BA, we recommend the introductory programming sequence and the introductory art sequence, deferring calculus to the second year. Either approach allows students to be on track for either the BA or the BS.

> **As time progresses, students naturally find their balance and self-select between the BA and BS degree options.**

As time progresses, students naturally find their balance and self-select between the two degree options. Within the BA option, students need to select a double major that best suits their interest: DMS, EMAD, or SA. We expect some students will find that computer science is not for them, drop the game development major, and move into DMS, EMAD, SA, or some other major. We also expect that some students will decide to switch into our computer science major.

Game development degree requirements

The University of Denver operates on a quarter system with most classes being four credits. One four-credit class on our system is roughly equivalent to a three-credit class on the semester system.

The bachelor of science degree has the following requirements:

- 48 credits (12 classes) in computer science; of these, 11 classes are required: computer science introduction I, II, and III; discrete structures; computer organization; operating systems; game programming I and II; graphics I; and senior project/portfolio;
- 20 credits (5 classes) in mathematics;
- 20 credits (5 classes) to serve as "cognate" in DMS, EMAD, or SA;
- 6 credits (14 classes) in general university foundations; and
- 39 elective credits.

The bachelor of arts degree has the following requirements:

- 44 credits (11 classes) in computer science; of these, nine classes are required: introduction to computer science I, II, and III; discrete structures; game programming I and II; graphics I; and senior project/portfolio;
- 4 credits (1 class) in digital media studies (animation);
- 8 credits (2 classes) in mathematics;
- 60 credits (15 classes) double major in DMS, EMAD, or SA; and
- 68 credits (17 classes) in general university foundations.

On the technical side, the major difference between the BS and BA is that the BS requires computer organization and operating systems whereas the BA does not, and the BS requires five mathematics classes whereas the BA only requires two. The other major difference between the two degrees is that the BA requires significantly more emphasis on DMS, EMAD, or SA.

COMPUTER SCIENCE TECHNICAL CURRICULA

Most required computer science classes follow formats and topics found in computer science departments nationwide. Hence, we only discuss those courses in our curriculum that are more directly related to games. Our program assumes that students take the introductory sequence, algorithms and data structures, and then game programming I, graphics I, game programming II, and then the senior project, in that order.

Introductory curriculum

We have begun modifying our freshman sequence to better serve and retain incoming majors. Prior to this year, many students viewed our introductory sequence as abstract and boring and hence too difficult. The sequence consisted of three classes in which the first two focused on introductory programming using C++ and the third covered the standard template library and object-oriented programming. The effect was to lose majors during the first year.

Some instructors feel the "sink or swim" attitude toward new majors is appropriate as it weeds out those who do not belong in CS. On the contrary, we believe that many students capable of finding rewarding computer science and related careers are unnecessarily discouraged and that a new approach is needed. The challenge is how to do this and still retain the same competency standards.

After much discussion and planning, we are trying a new freshman sequence this year in which the first class

uses 2D game programming in flash/ActionScript to teach introductory programming. The creation of games provides concrete examples, thus addressing concerns about abstract content. Because the elementary ActionScript 2.0 language is almost identical to C++, learned knowledge is directly transferable to C++. The second course switches to a more traditional C++ approach. We anticipate that prefacing the class with concrete examples will ease the jump to abstraction.

The third-quarter class is again concrete in that students work in teams in a project-based learning environment in which they develop a multistep tool that is graphical in some aspect. One example is to create a miniature flash-like drawing and animation tool using C++ and 2D openGL. In this class, students also learn UML, testing, programming using an API (openGL), and the concepts of tool building. The underlying goal of the third quarter is to solidify programming skills so that students are ready for data structures at the beginning of their sophomore year.

The first- and third-quarter courses have strong visual aspects, thus making them more engaging and helping students see the effect of abstract code. Our first experiment with using flash/ActionScript resulted in feedback indicating that most students felt seeing objects move on the screen in response to their code made learning the language much easier. All three quarters use pair programming, which has been shown to be helpful in retaining women in computer science.[1] Both traditional computer science majors and game development majors are taking this sequence together.

> **For a student planning to pursue a digital art career, a firm foundation in studio art is just as important as specific digital tools.**

Game Programming I

This first course in 3D game design uses a game engine, currently Torque, and scripting to provide students an overall game-creation experience. Topics include visual asset creation (3D modeling in Blender, Milkshape, Max, Maya, or Rhino), texturing, asset importing and placement, programming interactivity, score keeping, levels, terrain generation, and physics and particles. This includes understanding the game engine software architecture and object model in sufficient detail to use it effectively. Students work in teams to create a final game using the engine. Game Programming II covers how parts of an engine are implemented.

Graphics I

Taken by both CS and non-CS majors, including students from traditional degrees such as engineering and physics, this class is a general introduction to computer graphics. Topics covered include graphics hardware, graphics APIs, mathematics for computer graphics, viewing transformations, windows and viewports, fundamental algorithms (scan conversion, clipping, hidden surface removal, and so on), polygonal meshes, shading and illumination models, and various human-computer interface issues.

Game Programming II

This class teaches the fundamental algorithms and data structures used in the graphical and physics portions of a game engine. It assumes that the student is familiar with basic graphics and at least one game engine. Topics covered include texturing, scene representation, updating and rendering, levels of detail, terrain representation and generation, collision detection, and physics-based behavior and modeling.

Senior project/Portfolio

The senior project is intended to serve as a capstone class as well as help students prepare a portfolio. In this two-quarter sequence, students will work in teams of four to six to develop games including visual asset creation, sound, and coding. Teams will pitch the idea to faculty, provide design documents, and manage themselves to meet agreed-upon milestones. During this time, students will also work with faculty to assemble a portfolio demonstrating breadth as well as depth in one or more areas of interest to the student.

STUDIO ART CURRICULA

Studio art plays a fundamental role in game development. For a student planning to pursue a digital art career, a firm foundation in studio art is just as important as specific digital tools. Our SA program provides courses in drawing, painting, sculpture, ceramics, printmaking, and photography.

Introductory art sequence

Just as game development is an interdisciplinary field, so are the visual arts. To prepare students by giving them a broader articulation perspective, the CAMP program, a new freshmen sequence, was created this year with several primary intentions:

- provide a unique one-year art foundation experience and peer community,
- encourage a marriage of technical and conceptual skill building,
- emphasize a more connected or interdisciplinary approach to materials and design basics, and
- enable students to identify their own strengths and trajectory upon completing the program.

Most art programs at the college level provide a foundations sequence, many of which follow a model of 2D

design, 3D design, and drawing. CAMP is similar with respect to covering these basics, but differs in format and the inclusion of digital media. Rather than separating 2D from 3D and so on, CAMP aims to integrate these areas and increase the level of complexity as students move through the program—beginning with design basics and progressing to cultivate visual perceptions and analysis. Students rotate through the segments and have contact with different faculty in each segment.

Contemporary art practice and pedagogy rarely fall into strict categories—for example, drawing is no longer just graphite on paper. The program design seeks to break down media boundaries and expose students to multiple possibilities. In CAMP I, for example, particular attention is given to the connections between segments so a student understands a volume as a 2D, 3D, and digital entity.

This reinforcement of art elements and vocabulary continues throughout each segment, along with an effort to deny any preferential approach. No segment is given any more or less importance than the others, and students are expected to approach each segment with vigor, regardless of their previous experience with that media.

In addition, students are encouraged to apply a problem-solving approach while being exposed to this variety of media, and they are expected to develop a vocabulary to use in making and discussing various forms of visual communication. This is essential, given that freshmen students enter the School of Art and Art History with different and uneven backgrounds in art from their high school education. At its most simplistic, CAMP levels the playing field and brings all students to a more informed and even position.

> **Creating environments, characters, and simulated real-world objects is part of the plethora of tools essential to the game design student.**

CAMP strives to create a culture that empowers students and gives them basic yet varied technical and conceptual tools rather than a series of unrelated prerequisite courses. The CAMP faculty also stresses personal investment, exploration, and risk taking as students move through the program. Ultimately, students build a greater sense of self-awareness, increased agency as content makers, and a more flexible approach to art media that empowers them to self-select their potential areas of interest such as painting, sculpture, digital media, and so on.

General studio art classes

Studio art teaching methods have been refined over the course of centuries to encourage and reward both creative and critical thinking as well as visual refinement. These skills are transferable to all aspects of videogame production beyond the obvious needs of visual asset creation.

Drawing and studio art

A complex and multilayered activity combining visual, cognitive, and handicraft skills, drawing is a fundamental skill in the arts. In simple terms, drawing teaches students to represent what they see to others. Representational drawing requires students to perform cognitive translations from observed 3D reality onto 2D surfaces in real time. This high-speed decision-making is guided by aesthetic judgment, with the goal of creating a pleasing visual composition.

Drawing's importance to game development is manifold. Drawing is an imperative skill for communicating visual ideas to others. The observational skills necessary to render illusions of reality are first acquired and internalized through drawing. Loose drawings—sketches — are the method for rapid prototyping (RP) visual ideas for character design, settings, and point of view. Storyboarding comprises sequences of drawings that detail visual changes over time. Verisimilitude is the current highest standard of representation in games.

Physical/virtual sculpture

Sculpture offers a unique opportunity to tie together the virtual with the physical in a game development program. As humans grow and develop, they learn to use and depend on the sense of touch to gain knowledge and understanding of the world. This tactile relationship to the real world is essential to the formative comprehension of the physical.

Creating environments, characters, and simulated real-world objects is part of the plethora of tools essential to the game design student. While these are generally created in the virtual environment using modeling and rendering software packages, a fundamental component is sometimes absent from a real-world associative experience. The fact that this information is bound to a visual on a screen is in some way its own limitation for creative intervention.

The most direct, natural way we have for understanding virtual reality is first to transform it into the real physical object. Students can use RP equipment to achieve this. They can send computer-aided design image files to an RP machine, which creates physical models directly from digital data in hours. The observer can physically touch and manipulate these 3D representations, which convey information not obtainable from 2D projections.

Students can use a variety of means available in industry today to generate the CAD design files. They can create 3D objects within the computer with modeling software or, alternatively, they can sculpt the objects by hand and then scan them using 3D laser technology. Once in the digital realm, the designers can easily alter

or collage these files within the computer, permitting another avenue of creative intervention, and then send them to an RP machine for final output and analysis.

Whether a student is working on a project for a sculpture class or a game development class, creating prototypes and conceptual models is essential for exploring physical relationships. Having the ability to move between these physical and virtual options provides a unique learning environment that lets students further their knowledge of the real world through abstractions.

GAME DESIGN AND ANIMATION CURRICULA

We have taught seminar game design classes and are now in the process of articulating a studio class in a lecture and lab format. The objective will be to give students a framework within which to create and upon which to experiment and expand. We have seen some success with using obscure non-western board games to draw attention to the structures of rules that comprise game play. We look to texts by Chris Crawford[2] and by Katie Salen and Eric Zimmerman[3] for their lucid explanations of design principles. We are also on the lookout for other authors who might have contrasting views.

The animation and modeling sequence is aimed at giving students grounding in the foundations of 3D modeling, surfacing, rendering, and animation.

The field has so many opportunities that we don't want to stifle students by conveying a rigid message that says, "This is the right way to design a game." We focus on using modest technologies to complete modest games in the time available during class. We are in the process of adopting and adapting the CMU Entertainment Technology Center approach of designing and building small prototypes of numerous games in a short period.

It is tempting to create courses explicitly in humane-game design, but doing so limits the broader space of games and how humane concerns fit in. Our current approach is to suffuse the humane focus into the curriculum in a natural way through the teaching and interaction with faculty, all of whom embrace the humane-game model.

Animation and modeling

The third animation and modeling sequence consists of three courses, Introduction to 3D Modeling and Animation, Character Animation Principles, and Short Animated Film Production, which are taught through the DMS program. The sequence is aimed at giving students grounding in the foundations of 3D modeling, surfacing, rendering, and animation. As the sequence progresses, students work on more complex animations that are informed by the various principles of traditional animation and culminate in the production of a short animated film.

Introductory course. The introductory course provides a brief look at the techniques and processes involved with each of these key disciplines. The modeling section of the course introduces polygonal modeling, spline-based approaches, and subdivision surfaces. Students progress to work with surfaces and textures on their objects. The course explores both algorithmic and bitmap textures as well as complex layered surfacing techniques. Virtual lighting and rendering methods are discussed and employed in the class projects, and the basics of global illumination are examined.

The final portion of the class is devoted to simple animation principles, including keyframing and inbetweening, as well as a brief introduction to character animation principles such as squash and stretch, slow in and out, and overlapping action. Students are introduced to a variety of processes for creating an animation. Later classes emphasize and build upon the production workflow of planning, blocking, and polishing.

Character animation. The character animation class is primarily concerned with the 12 fundamental character animation principles that originated with the Golden Age Disney animators.[4] The students work on a variety of projects and exercises to build a firm working knowledge of these principles. Projects cover secondary actions related to moving holds, walk cycles, and basic acting as it applies to animated characters.

The course employs preexisting characters of increasing complexity and sophistication as well as student-created characters. In addition to focusing upon the animation principles, the course introduces the basics of rigging and skinning characters. These technical direction skills support a level of self-sufficiency that is capitalized upon in the final course.

Film production. The final course in the sequence concentrates on the preproduction and production of a short animated piece. Topics covered in the film production class include story development, cinematography, and acting choices. Some time is also spent on using particle and dynamics systems for effects.

The class allows students to work in small groups to produce a finished piece of animation that demonstrates their skills and capabilities. The class begins with story development, brainstorming, and storyboarding of their ideas. These ideas are polished, and more detailed planning is used to create a clear plan of action for the animation.

The students use their plans and storyboards to create animatics to further develop their framing, timing, and layout skills. They convert these animatics into 3D layouts that provide a foundation for the finished animation. The students spend the majority of the class

creating models and rigging and animating characters for the finished animation.

IMPORTANT PROGRAM CHOICES

Creating a new program provides opportunities for setting direction but also requires some important and sometimes difficult choices. Two key choices were made while creating this program:

- *Establishing a humane-gaming focus.* To attract more students, the temptation is to just give students what they want. Many males, arguably the major recruiting target for game programs, play adventure and first-person shooter games. Students hope to make games in a genre they love. Humane games may not be the first choice of some of our students, yet we feel it is important to expose them to this area. We believe this exposure will result in better game programmers and designers within first-person and adventure games as well as promoting new game applications and genres.

> **Any person working in the game field needs some appreciation of art, in particular the critique process.**

- *Requiring technical students to take art.* We require all game development majors to take at least five SA, EMAD, or DMS classes including at least the first two SA foundation classes. This is perhaps our most controversial requirement. Game development requires artists working with programmers. We feel that any person working in the game field needs some appreciation of art, in particular the critique process. This appreciation will help them to better understand and communicate with their artist coworkers. We also believe in whole-brain education and hope that many of our students will find they have abilities in the creative side as well as the technical and will either choose the BA double major or at least choose some additional art electives.

OBSTACLES IN CREATING A GAME DEVELOPMENT DEGREE

Getting a new major approved at a university can be a political minefield, especially when that major has the word *games* in the title. The following were obstacles in getting approval of the new major:

- *Perception that this is not an academic subject.* Many people initially balked at the idea of a major in creating games, mistakenly thinking it was just a trade-school-type education. They did not understand the depth of computer science, physics, design, and art knowledge needed in this field. Once the required knowledge was explained to the various approval committees, emphasizing how this focus ties in with traditional computer science, digital media, and art, most realized this truly is an academic subject.
- *Ignorance of the potential of games.* Again, many adults seem to think of games as merely a diversion for children, one that interferes with what they should be doing. Although this claim may sometimes be true, such objections misdirect the focus from the many current and upcoming applications of games that make this such an important field. Many people are also surprised to hear that the entertainment game industry is now as big a business as Hollywood.
- *Concerns about reputation.* Recent negative press about the sexual and violence content of games seems to have fueled the concern about reputation. People questioned whether a university wants to be associated with games. Once educated about trends such as the fact that 93 percent of games are rated "teen" or "everyone," and once we explained our humane-games focus, various approval committees became less reluctant to support the program.

The following are ongoing obstacles in program building:

- *Parents.* Many parents seem to share the view that game development is not an academic subject. We have met with dozens of parents who have reluctantly come with their child to visit our university. They seem to be resigned to the visit only because it is what their child wants to do. Once they understand the program, most parents become as excited as their children. The challenge here is to reach out to these parents and educate them about the potential and academic merit of game development. Perhaps this parental education will follow automatically as the field matures and more institutions create game-related programs.
- *Normal academic barriers to collaboration such as budgets, tuition flow, and joint hiring.* At many universities, art and computer science are not housed within the same academic division, thus adding to the obstacles. We are overcoming these obstacles at our institution, but they will likely be a concern for any school wishing to develop a new collaborative program such as ours.

PROGRAM GOALS AND HUMANE GAMING

One of the goals when creating our program was to rebuild the number of undergraduate computer science majors at DU. Our first cohort of game development students started in fall 2005 with 10 incoming majors. During fall 2005 and the first two months of 2006, we

met with more than 40 students who came to tour the campus and discuss our new game development major. Based on historic computer science yields from this type of student visit, we expect 30 new game development majors in fall 2006. DU is a relatively small school that enrolls about 4,000 undergraduate students and 5,000 graduate students. Thus, this increase in game development majors is significant.

A related goal is to increase the number of women and underrepresented minorities in game development and computer science. The current game development workforce comprises less than 11 percent women and 4.5 percent African Americans and Latin Americans.[5] Games have a significant impact on our younger generations, yet they are being created by a predominantly white male workforce. We believe that a focus on humane gaming in college programs could help to attract more women and underrepresented minorities to the field, thus giving these populations more of a voice in shaping this important industry.

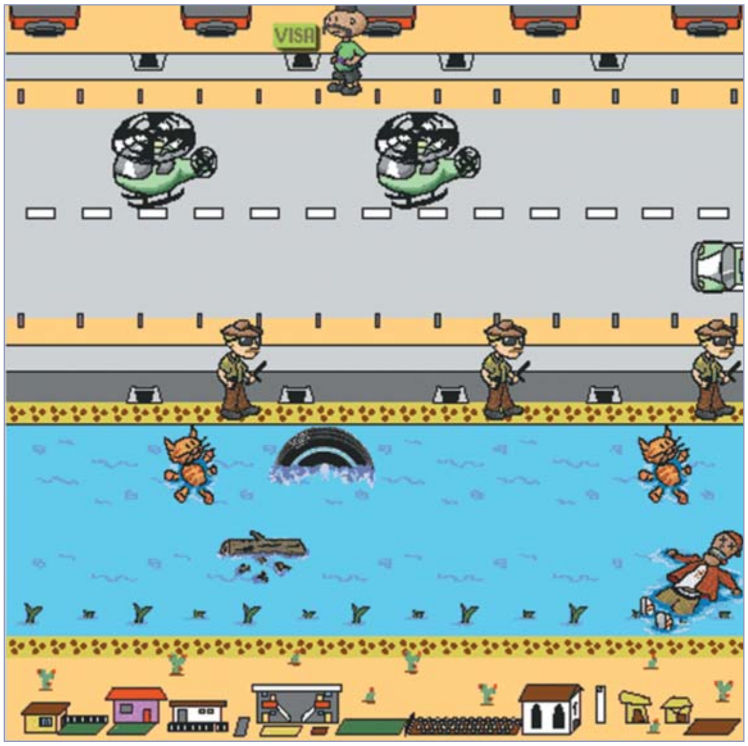

Figure 1.* Crosser,[5] *an example of a socially conscious game, deals with border immigration issues, raising awareness of the situation through game play.

Finally, our program aims to direct this newly developing academic field toward benevolent applications in what we call humane gaming. We envision humane gaming to consist of three components:

- *Educational games*. Educational games, interactive simulations, and training tools are applicable to almost every field and academic discipline. Creating these games requires significant computer science and design experience as well as domain knowledge. In addition to using educational games to teach domain knowledge, the process of creating games can be used to teach mathematics, computer science, physics, art, and literacy in an integrated fashion. We are working with the DU School of Education to build a computer science, mathematics, and game certification program for high school teachers.
- *Medical games*. Games are being used in medicine for mitigating pain, to promote exercise, to reduce stress, to rehabilitate patients with damaged cognitive abilities, and to help children feel empowered to recover from chronic illnesses such as cancer. Although many of these applications require medical domain expertise, we will expose students to these game applications and advise them about graduate degrees in medicine, psychology, and cognitive science.
- *Socially conscious games*. Undergraduate students in our programs will be able to participate in creating socially conscious games starting from the first quarter. The goal of these games is to raise social awareness or advocate for a cause.[6] Examples include *Crosser*, *Steer Madness, Newsgaming.com, Maria Sisters, Peacemaker, Rethinking Wargames Activate!*, and *Escape from Woomera*. As an example, *Crosser*, shown in Figure 1, deals with border immigration issues, raising awareness of the situation through game play.

As academics participating in the birth of a new discipline that has significant social impact, we have a stewardship responsibility and an opportunity to explore benevolent game applications as well as to make game development more inclusive of women and underrepresented populations.

References

1. L. Werner et al., “Want to Increase Retention of Your Female Students?” *Computing Research News*, vol. 17, no. 2, 2005; www.cra.org/CRN/articles/march05/werner.html.
2. C. Crawford, *Chris Crawford on Game Design*, New Riders Games, 2003.
3. K. Salen and E. Zimmerman, *Rules of Play: Game Design Fundamentals*, MIT Press, 2003.
4. O. Johnston and F. Thomas, *The Illusion of Life: Disney Animation*, Disney Editions, 1995.

5. A. Gourdin, "Game Developer Demographics: An Exploration of Workforce Diversity," Oct. 2005; www.igda.org/diversity/IGDA_DeveloperDemographics_Oct05.pdf.
6. R. Fajardo, "Pixels, Politics and Play: Digital Video Games as Social Commentary," Intelligent Agent, vol. 3, no. 2, 2003; www.intelligentagent.com/archive/Vol3_No2_gaming_fajardo.html.

Lawrence Argent *is a professor at the University of Denver. His research interests focus on ideological exploration involving aspects of technology's translation of imagery/form and the acceptance and assumptive cognition of such information/abstraction as a new reality. Argent received an MFA in sculpture from the Rinehart School of Sculpture at the Maryland Institute, College of Art. Contact him at largent@du.edu; www.lawrenceargent.com.*

Bill Depper *is an assistant professor at the University of Denver. His research interests include animation and game development. He received an MFA in creative writing from the University of Iowa. Contact him at wdepper@du.edu.*

Rafael Fajardo *is an associate professor at the University of Denver. His research interests include the expressive potential of video games, especially their ability to work as cultural commentary. Fajardo received an MFA in graphic design from the Rhode Island School of Design. Contact him at rfajardo@du.edu.*

Sarah Gjertson *is an assistant professor at the University of Denver. Her research interests and studio work explore nostalgia, issues surrounding social and personal comfort, and consumption within American popular culture. She received an MFA from the School of the Art Institute of Chicago. Contact her at sgjertso@du.edu.*

Scott T. Leutenegger *is a professor and founder and director of the Game Development Program at the University of Denver. His research interests include spatial databases, humane gaming, and computer science education. Leutenegger received a PhD in computer science from the University of Wisconsin. He is a member of the ACM and the IEEE. Contact him at leut@cs.du.edu; www.cs.du.edu/~leut.*

Mario A. Lopez *is a professor of computer science at the University of Denver. His research interests include computational geometry and spatial databases. Lopez received a PhD in computer science from the University of Minnesota. Contact him mlopez@cs.du.edu.*

Jeff Rutenbeck *is an associate professor and founder and director of Digital Media Studies at the University of Denver. His research interests include media history, media theory, social implications of emerging technologies, and hypercomplexity. Rutenbeck received a PhD in communication from the University of Washington. He is the founding president and current chairman of the board of the International Digital Media and Arts Association. Contact him at jrutenbe@du.edu.*