

Computer Graphics: Problem Based Learning and Interactive Embodied Pedagogical Agents

F. Seron, E. Cerezo and S. Baldassarri

Advanced Computer Graphics Group (GIGA)
Computer Science Department, Engineering Research Institute of Aragon (I3A)
University of Zaragoza, Spain

Abstract

This paper focuses on the use of new tools in order to improve the learning of Computer Graphics, in particular of some aspects that have been detected as specially difficult by the students. Within this context, this paper explores the union of two areas: the educational strategy named Problem-Based Learning (PBL) and the use of interactive embodied pedagogical agents (IEPAs) for education and training tasks. By combining these ideas, we obtain an interactive learning environment created to improve student's learning capabilities. From the PBL point of view, the real problem proposed to the students consists in the development of their own viewing system. While for introducing IEPAs we've developed a powerful engine, called Maxine, that enables the management of 3D environments with special emphasis in the use of embodied animated agents. The engine, based on open source libraries, allows emotional and multimodal interaction with the user in real-time: via text, voice, images, animation (facial expression, lip-synch, body gestures) and the choice of the answers. By carefully orchestrating both ideas the students are encouraged to care more about their own progress, convey the enthusiasm in the learner for the subject matter and simply make learning more fun. Evaluations already done to the students show promising results.

Categories and Subject Descriptors (according to ACM CCS): I.3.8 [Computer Graphics]: Applications

1. Introduction

This paper presents the experience of teaching Computer Graphics (CG) subjects in the Computer Science degree. Through the years, we detected that CG students encountered several problems to assimilate certain fundamental concepts; the difficulties came, in most cases, from the evident initial difficulty for the student to visually imagine concepts.

In order to tackle these problems, we decided to explore the union of two areas. The first area corresponds to the educational strategy named Problem-Based Learning (PBL). The second area is the use of embodied pedagogical agents in education and training tasks. By combining these ideas, we arrive at the birth of an interactive learning environment created to improve student's learning capabilities. The use of the PBL methodology during the courses has been of great success. The student is asked to develop a graphics system,

very simple in its beginnings but with increasing functionalities added as the courses progress. The fact that the student finally achieves to create a system that offers similar features to many commercial ones, constitutes a big motivation for the student. Even though many of the problems are cleared as the students develop their own graphics system, it would be of great interest to speed up their comprehension. Looking for a way that allow students to "play" and work with the problems they find, we considered the use of interactive embodied agents. The engine beyond the agents allows to present some concepts or applications (presentations), to interact with the students (tutorials) and also can be used itself by asking the students to apply Reverse Engineering to the system. The proposed learning-process is highly interactive and flexible, in contrast to common present-day passive methods and perfectly fits current tendencies linked to European Bologna process, the life long learning and the more and more used "e-learning courses".

This paper is organized as follows. Section 2 presents the academic environment and the difficulties detected by the students in the learning of Computer Graphics Subjects. Section 3 presents the use of the PBL methodology while Section 4 introduces the incorporation of Interactive Embodied Pedagogical Agents and the new teaching activities. In Section 5 the evaluation of the students' satisfaction using these tools is discussed. And finally, in Section 6, some conclusions and future work are commented.

2. Teaching Computer Graphics subjects

2.1. Academic Environment

The teaching activity focuses basically on the set of subjects collectively known as Computer Graphics. The first subject, Computer Graphics (GG), describes the basis of synthetic image generation. The second, Geometric Modeling (GM), determines the shape of objects by means of mathematical and algorithmic model-description, and finally, Visual Modeling and Animation (VMA) is concerned with the visual appearance of objects and their kinematics and dynamics. These three subjects are respectively taught in the sixth, seventh and ninth semesters of the Computer Science degree. The degree is a ten-semester (5-year) study at the University of Zaragoza (Spain). Each subject is an optional course of 60 hours, distributed in 30 hours of theoretical sessions, 15 of problem solving sessions and 15 of practice sessions in a computer lab. The contents of the Computer Graphics courses are detailed in Figure 1. In order to explain the student's desired degree of expertise, a key follows every subject. This key has been labeled as follows:

- I: Informative level. The student should be aware of the general ideas concerning the subject and handle related software with fluency.
- A: Advanced level. Student should be able to read and understand a textbook related with the subject, and design and implement applications that carry on that issue.
- E: Expert level. Student should be able to read and understand an expert level paper related with the subject, to design and implement applications that carry on that issue and, in some cases, propose new or improve methods or algorithms.

2.2. Teaching and learning the most difficult topics

The most important difficulties encountered in the transmission of CG knowledge may be classified in two groups: those that are due to a lack of scientific basis in the previous curriculum and those that are due to inherent conceptual difficulty or the level of knowledge demanded.

The lack of previous knowledge of more than elementary Geometry forces the explanation of such knowledge from the outset. Therefore, part of the CG curricula and the greater part of GM are devoted to these issues. On the other hand

this is also advantageous, because the concepts are introduced in a way which is totally geared to their application in Computer Graphics. In CG the Geometric Transformations and Geometric Projections are part of this group. And in GM they are the following: Basic concepts of Analytic Geometry, Differential Geometry and Interpolation and Approximation Geometric Modeling of Euclid Objects (Curves and Surfaces). The case is the same for knowledge related to Physics (Optics and Mechanics), in particular the following concepts, which are imparted in VMA: The Rendering Equation, Algorithms for Shading (Radiosity, Ray Tracing, Radiance Techniques), Colorimetry (Color Representation, Color Models) and Animation (Interpolation, Direct and Inverse Kinematics, Restrictions, Direct and Inverse Dynamics).

According to the students' opinion, the most difficult concepts to understand are usually (reasons included):

- In CG: Geometric Projections and 3D Viewing Systems. Problems are due to the complexity of array treatment of transformations and projections.
- In GM: Basic concepts of Differential Geometry. Mathematical treatment is totally novel. Geometric Modeling of Euclid Objects (Curves and Surfaces), problems are located in the comprehension of parametric representations.
- In VMA: Global shading. Direct and inverse kinematics. Virtual humans. Facial animation. Behavior. In all the cases the major difficulty lies in the technical complexity of its implementation.

A problem common to all these subjects consists in the evident initial difficulty for the student to visualise certain concepts. In order to solve most of the student's doubts and problems the use of the PBL methodology and IEPAs are introduced in the courses.

3. The PBL methodology

The main objective in CG courses is to supply the student with the knowledge and abilities needed to develop a simple but complete graphic system. This system must have some basic features in the first version, and evolves continuously by incorporating the concepts that are consecutively explained during the three subjects that comprise the course. The teaching methodology is rooted in Problem-solving Based Learning (PBL); it is a learning strategy that encourages students to solve real problems. This methodology focuses on the student both as an individual and as a member of a working group and considers learning as a communication process [GS92]. In PBL the problem or project is the milestone of the whole learning process, is the way in which students learn different skills. Students are in charge of their own advance and lecturers must provide the necessary tools to make the students' work possible. The objectives of the PBL methodology are [Sav00]: integration of knowledge and skills, work and learning autonomy, teamwork, self-assessment, development of high level intellectual

COMPUTER GRAPHICS	GEOMETRIC MODELING	VISUAL MODELING AND ANIMATION
Introduction (I) Presentation Human Perception and Computer Graphics	Introduction (I) Presentation Historic Evolution	Introduction (I) Presentation Historic Evolution
Historic Evolution (I)	Basic concepts of (I) Analytic Geometry	Local Shading Models (A) The Local Equation
Graphic Hardware (I) Output and Input Devices Graphic Accelerator Cards	Differential Geometry Interpolation, Approximation	Calculus algorithms Advanced Effects: texturing, roughness
Basic ideas (A) Scan converting lines Polygonal Objects Modeling Elements involved in Image Definition	Geom. Mod. of Euclid Objects: <i>Curves</i> (A) Plane Curves: Parametric and not parametric curves Space Curves: Spline, Bézier, B-spline, Fitting, Subdivision Parabolic Blending	Global Shading Models (A) The Rendering Equation Algorithms for Shading Radiosity Ray Tracing Radiance Techniques
Geometric Transformations (E) 2D Transformations 3D Transformations	<i>Surfaces</i> (A) Surfaces of Revolution Sweep and Quadric Surfaces	Colorimetry (I) Color Representation Color Models
Geometric Projections (E) Planar Geometric Projections Parallel and Perspective Projections	Bilinear, Ruled, Developable and Coons surfaces Bézier and B-spline Surfaces. Fitting and Subdivision.	Basic Animation Techniques (A) Key-framing Interpolation, blending Motion graphs Direct and Inverse Kinematics.
3D Viewing Systems (E)	<i>Solids</i> (I)	Advanced Anim. Techniques (A) Dynamics Facial animation Motion capture Particle animation Virtual humans Behavior Techniques
Rendering (A) Face and Surface Culling Clipping Rasterization Basic Illumination Model (I)	Geom. Mod. of Fractal Obj. (I) Deterministic Random Implementation Techniques Direct Procedural	
Graphic Standards (A) Open-GL		
Simple scene generation (I)	Complex scene generation (I)	

Figure 1: Computer Graphics subject contents and the desired students degree of expertise

activities (analysis, synthesis and evaluation), and argumentation and critical reasoning. Problem solving is a process that requires the knowledge of a major discipline and of the techniques and skills needed to bridge the gap between the problem and the solution. Nowadays, the PBL methodology is used in Higher Education to impart classes in different areas such as electronics [CHL07], biomedical engineering [AR07], physics [vBK*04], mobile applications [MRK06], and, also in Computer Graphics subjects [MGJ06].

In our environment, the initial problem, proposed in the first subject, consists in the "development of a viewing system that must satisfy certain minimal features". The user specifies a view or eye point and a view plane distance. A viewing direction or view plane normal is established by the line from the view point to the world coordinate origin. This constraint is fine for single object scenes, but it is an unworkable constraint in the case of multiple object scenes. Anyway, this first viewing system is easy to implement and understand and it is useful to gain experience by means of manipulating the different parameters of the system. The next step is to develop a more elaborate system where the camera can point at any direction: the viewing direction is no longer constrained to point at the origin of the world coordinate system. The camera can also be rotated about the view plane normal by specifying a twist angle; these are the usual degrees of freedom that a real photographer would use to position a real camera. The third step is to develop a more sophisticated viewing system in which a viewing volume is in-

troduced. To define this view volume it is necessary to specify a near plane, a far plane and a view plane window. In all three versions of the system, the modeling technique of the geometric objects is the classic representational form based on a mesh of polygonal facets, and the renderers developed must be a wireframe projection and a complete local shading model using a Z-buffer and interpolative shading techniques (Gouraud or Phong).

In the second subject, "the system must increase the ability to model geometry". The transition from polygon meshes to patch meshes is natural: if a mesh of four-sided polygons is considered as an approximation of a curve surface, then a parametric patch mesh can be considered to be a set of curvilinear polygons which actually lie in the surface. The aim of the students' project is now to develop a surface generation and manipulation program. Any of the surfaces discussed in the class, e.g. bilinear, ruled, linear or bicubic Coons, Bézier or B-spline surfaces, both non-rational and rational, can be used as the basis of the project. The ability to generate and display varied surfaces makes investigation of the effects of input data variations particularly interesting. In order to display the results the students can use their previously developed viewing system. The best approach to render parametric patches is to pre-process the representation converting the patches into planar polygons. This results in low code complexity but, of course, certain inevitable problems arise in this simple approach.

In the third and last subject, "the system must increase the performance of the shading model employed". Texture mapping is one of the techniques that can be used to enhance the visual interest of a scene, and it may be added to a standard rendering method without an excessive increase of the processing cost. Students can use texture mapping with their polygon mesh representation. There are many ways to perform texture mapping. The choice of a particular method would mainly depend on time and quality constraints. Other possibilities students may use are global shading models such as ray tracing and radiosity. In these cases the purpose is to generate a basic implementation compatible with the data structure of the first viewing system developed.

The development of this graphics system of increasing difficulty through the three courses forces the student to use most of the concepts explained in the theoretical part of each subject. From a global point of view, the problem posed can be considered an "open problem" because it allows a variable number of solutions (a different one for each working group, even one for each student).

Regarding the assessment of students' performance, emphasis is placed on the comprehension of the underlying concepts and on the system capabilities, such as the completeness of the system, the quality of the scenes developed, the complexity of the models used, implementation efficiency, etc. The tutor evaluates the student by means of technical reports and the presentation of the team work, as well as the student's individual work and attitude during supervised periodical meetings.

However, we detected that most of the students still encounter several problems to assimilate certain fundamental concepts and the main difficulties are related with the mental visualization of that concepts. Even though many of these problems are cleared as the students develop their own graphics system, it would be of great interest to speed up their comprehension. To do so, the use of interactive embodied pedagogical agents has been included in the subjects.

4. Introducing Interactive Embodied Pedagogical Agents in the Computers Graphics subjects

The idea of using pedagogical agents comes from the inherent psychosocial nature of student-agent interactions and of humans' tendency to anthropomorphize software [RN98]. Recent evidence suggests that learning environments with lifelike characters can be pedagogically effective [LCS*97] while at the same time have a motivating effect on students [LCK*97], [ERL99], [LRL00]. Basically, interactive Embodied Pedagogical Agents consist in:

- "Agent" means that it is semi-autonomous.
- "Interactive" indicates that it can detect external stimuli such as keyboard, mouse position or clicks, voice...
- "Embodied" refers to its physical representation: it has

face and body, uses gestures to communicate, and can move around in a virtual scenario.

- "Pedagogical" implies that it is designed to help teaching.

The use of this kind of agents is an innovative area in teaching and, from the pedagogical point of view, agents may make learning more fun and motivate student to spend more time in the learning environment. Also, interactive embodied pedagogical agents offer a low-pressure learning environment that allows users to gain knowledge at their own pace. In this way, we have developed a script-directed engine, called Maxine, that makes it easy the use of an autonomous interactive embodied pedagogical agent as it is explained in the next section.

4.1. The system behind: Maxine and its agents

Maxine is a script-directed engine for the management and visualization of 3D virtual worlds. In Maxine it is possible to load models, animations, textures, sounds, etc., in real-time, as they are needed in the virtual environment. It has been written in C++, employing a set of open source libraries (as it is explained in [BCS07]). A big effort has been invested in the integration and communication of all these libraries. In any case, we succeeded in maintaining a good real-time performance (see Table 1).

The engine manages scene graphs that can be built in real-time, dynamically creating and manipulating its elements by means of a simple command interface. These commands can be executed via script-files when initiating the application or during execution, or can be introduced through the text console every time. The scripting language used is Lua. A scene graph can be represented by simple objects, like images, texts, videos, geometric primitives models, lights or 3D sound. But also by animated characters, with different types of animations including secondary animation to increase the expressivity and realism; animated actors, provided with facial animation (see Figure 2), synthetic voices with voice modulation for gaining expressivity and lip-synch. The engine can also manage several auxiliary elements like cameras, group of elements, animators (for animating group of elements),... and can include animations coming from motion capture systems.

In Maxine, virtual agents are endowed with the following differentiating features (for more details see [CBS07]):

- they support interaction with the user through different channels: text, voice (through natural language), peripherals (mouse, keyboard)
- they gather additional information on the user and the environment: noise level in the room, image-based estimate of the user's emotional state, etc.
- they have their own emotional state, which may vary during the interaction with the user and which modulates the agent's facial expressions, answers and voice.

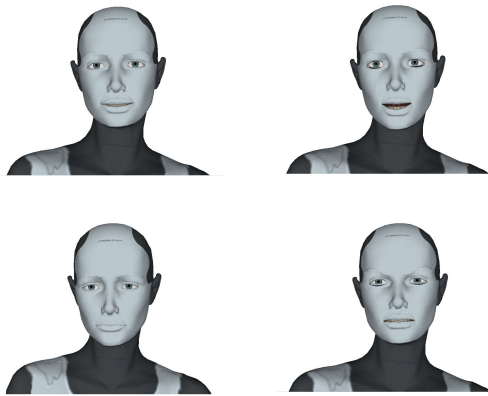


Figure 2: Different agent's facial expressions (clockwise: happiness, surprise, sadness, anger)

Table 1: Maxine real-time performance (frames per second figures correspond to full screen displays)

Scene	Number Vertices	Frame rate (fps)
Max actor	9176	90
Maxine actress	16667	31
Interactive Tutorials scene	20570	103
Demo group scene	75028	99
Rome colosseum scene	455851	12

4.2. New teaching activities

The system previously outlined enables us to undertake three types of new teaching activities: virtual presentations, interactive tutorials and reverse engineering exercises. Details about them follow.

A) **3D Virtual Presentations**, with the agent acting as an Expert Coach. Information presentation is a common and necessary educational act in the Information Society we live and work within: in our group we often make guided presentations to inform, teach, motivate and attract students, showing Computer Graphics applications and capabilities. In Figure 3 the agent is presenting the R&D activity of the group.

Agent capabilities allow to guide a student's attention with the most common and natural methods, gaze and deictic gesture: Maxine agents look at an object as point at it, can move through their environment, pointing at objects when discussing them (see Figure 3), look at the students when speaking to them... Effort has been invested in providing the virtual agent with facial and body expressiveness. Before and after these presentations the students have to answer questionnaires in order to know the impact of the agent in the students, not only from the visual point of view, but about the learning usefulness of the experience.



Figure 3: An IEPA points at an image while presents the R&D activities of our group

B) **Interactive tutorials** are used in order to make it easier to the student to understand the most difficult CG topics. One of the important problems encountered in the transmission of knowledge is the evident initial difficulty of imagining concepts visually. In this case, the Maxine system allows a better comprehension of the CG concepts. Its most novel contribution is a collection of interactive examples that can be manipulated by the teacher during the "theoretical explanations" by modifying the parameters related to the concept and visualizing the effects in real-time. Students may also use these examples in the lab, in particular, students can (see Figure 4):

- create and modify objects using classical primitives: points, curves and surfaces, based in a parametrical representation
- apply geometric transformation to objects
- modify its visual properties
- put and modify lights and cameras
- understand the scene organization through the visualization and use of a typical scenegraph structure
- analyse the 3D viewing system of Maxine
- modify the kind of projection
- load animations and use different blending alternatives
- create simple actors and change their expressions
- associate voices to the actors and make them speak
- provide the actors with secondary animations

All these aspects have to do with most of the concepts that the students point out as the most difficult topics. Unfortunately, till the moment, the aspects related with global shading can't be tackle with Maxine's features. But they can be approached by the "viewing system" selected by the student.

In addition to verbal feedback, Maxine agents can also use nonverbal communication to influence the student. For example, nodding or smiling to indicate approval, shaking the head to indicate disapproval, or presenting a look of puzzlement after a student error, or a pleasant surprise when the student finishes a task.

It is not difficult for the teacher to develop or modify the tutorials by means of the scripting language used to control

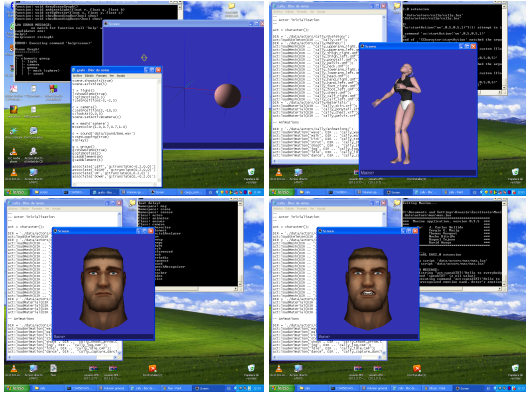


Figure 4: "Playing" with Maxine engine: managing a 3D scene (above left), loading and blending animations (above right) and changing characters' expressions (below)

every aspect of Maxine, as it has been briefly commented in previous subsection.

C) Reverse engineering, to teach all the technical issues behind the engine, with the agent acting as an Expert Practice Partner. We use Reverse Engineering (RE), the process of discovering the technological principles of a device/object or system through analysis of its structure, function and operation. In our case, and after having used it, students have to understand the specification of Maxine system without looking into its implementation details, just using the agent to show and test the fundamentals of the system architecture. This use is only recently and partly being done, but results seem promising.

5. Evaluation

Concerning with the evaluation of the use of PBL and IEPAs in the Computer Graphics subjects, we have considered several aspects.

With regard to the level of acceptance of the methodology employed, we considered students' questionnaires. The average values of the students assessment were compared with the remaining subjects taught by the Computing Area (40 different subjects), with the remaining subjects imparted by the Department (62 teachers and two areas) and with all the subjects imparted in our Engineering School (4500 students, 15 departments, 30 areas, 335 teachers). In all the cases the results of the Computer Graphics subjects get the best marks.

The analysis of the academic results of the students in the CG subjects shows that the percentage of students who passed is around 65%. And, moreover, 70% of the students receive high scores (A or B).

In order to know whether the use of virtual agents improves or not the learning outcomes and the students' perceptions about these agents, we have collected information

from the students. As we mentioned before, we have evaluated the virtual presentations and the interactive tutorials, since the use of Maxine in Reverse Engineering exercises is too recently.

Regarding virtual presentations, two questionnaires, one before and one after, are done. Their objectives are:

- To evaluate the previous knowledge of the subject that will be presented
- To assess the effectiveness of the "information provision" aspect of the message, ie, the Maxine's effect on the presentation subjects' comprehension
- To measure the perceptions about the Maxine agent.

Specifically, an introductory presentation about CG and its applications has been evaluated. The students are asked to evaluate their knowledge about different topics of CG before and after the presentation, evaluating it from 1 (very low) to 10 (very deep). In Figure 5 mean values are shown (classified by gender). It is interesting to realize that female students systematically rate their knowledge lower than male students; explanation of this behavior cannot be based on objective facts, as they all are in the same university level, having coursed almost the same subjects and having female students usually higher marks.

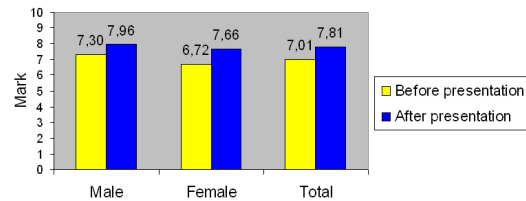


Figure 5: Effectiveness on the subjects' comprehension after a Maxine's presentation

Students are also asked about the aspects of the virtual agent that have attracted their attention (results in Figure 6) and which attributes would they use to describe the presenter (see Figure 7). About their perception of the virtual presen-

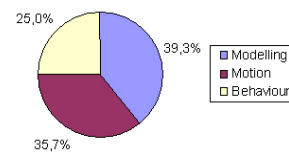


Figure 6: Remarkable aspects of the virtual presenter

ter as a good example of CG application 100% agreed, and 92% consider that the knowledge of the internal structure of the software behind the presenter would be a good way to acquire and teach CG concepts. However, most of them think that this kind of "virtual teachers" only can be used as a tool

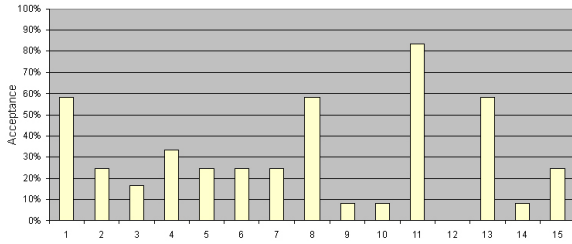


Figure 7: Acceptance of the following virtual presenter's attributes (1-Helpful, 2-Intelligent, 3-Likable, 4-Reliable, 5-Believable, 6-Competent, 7-Friendly, 8-Clear, 9-Natural, 10-Not very convincing, 11-Stiff, 12-Happy, 13-Neutral, 14-Sad, 15-Coherence expression-message)

Table 2: Helping to understand the most difficult topics

STATEMENT	RANK (1-10)
The use of the Maxine engine has helped my understanding of certain practical IG concepts.	8,0
The use of the Maxine engine has improved my understanding of IG theoretical concepts.	6,5
It would be interesting to use the internal structure of Maxine as a support to teach IG concepts.	8,8
The tutorial is easy to follow and comprehend.	8,0

and can not replace human tutors (75%), but could be a good option for distance training (92%).

Regarding the interactive session in the lab (typically lasting 2 hours), students agree that the use of embodied agents helped them to understand specific topics of CG (see Table 2). Moreover, they remark the utility of this kind of system to modify and visualize virtual environments and to allow the interaction with the agents. They are asked through an open question to specify which difficulties have they overcome thanks to Maxine. They mention that their understanding about coordinate systems, geometric transformations, perspective, and scenes creation have improved.

Results about the description of the virtual actor in the interactive session are shown in Table 3. By an open question students are also asked to compare Maxine characters with other virtual character they know (from videogames, programs,...). Their answers are all positive, considering them good, simple but effective. Students are also asked to point out which aspects contribute most to realism and to the lack of realism of the virtual agent. The answers have been divided into two groups: those corresponding to students being used to videogames and those that are not. It is specially interesting the different consideration about the aspects contributing most to the lack of realism (see Figure 8).

At the end of the questionnaire, two open questions are posed about which aspects of the work with Maxine were considered the most and the less interesting. Facial expres-

Table 3: Description of the virtual actor

STATEMENT	RANK (1-10)
The virtual actor looks real	7,7
The movements of his/her head look natural	7,7
His/her gaze looks natural	7,0
His/her facial expressions looks representative	8,0
Good lip-synchronization is achieved	7,5
Voice modulation is always coherent	6,6
Coherence between facial expression and voice	6,9

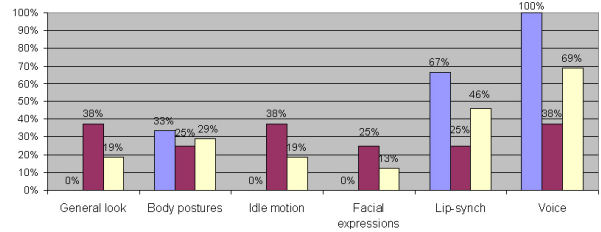


Figure 8: Aspects that contribute to the lack of realism. Opinion of students that: usually don't play videogames (blue), usually play videogames (red), total (yellow)

sions, quick actor response, emotions and the easiness of creating and manipulating a 3D scenario where stated as the most interesting topics. The most repeated complaint was about the use of a command interface. Nevertheless, usability is evaluated positively by the students, with all the marks between 6 and 9 (being 10 the highest). In particular, the highest marks correspond to the statements "I find the system easy to use", "I have enjoyed working with the system" and "I have felt comfortable working with the system". The lowest mark is dedicated to the statement "The interface is intuitive and user-friendly", but this is understandable since most of the commands regarding the animation and control of the virtual actors is done through a console by using the Maxine scripting language.

6. Conclusions

In this paper the teaching methodology used to solve specific difficulties detected in the learning of some concepts of Computer Graphics is reported. The teaching methodology is rooted in "Problem-solving Based Learning". The main objective is to supply the student with the knowledge and abilities needed to participate in the development of a simple but complete graphic system on his or her own. The system must have some basic features in the first version, and evolves continuously by incorporating the concepts that are consecutively explained during the three subjects that comprise the course. This process forces them to modify the point of view of their solutions as the technical possibilities change as they progress in their studies and elaborate their own knowledge. Success in this area allows the student

group to describe, interpret and justify all the components that constitute "their" own viewing system. In this way, the learning-process becomes highly interactive and flexible, in contrast to the more frequent present-day passive methods. The fact that the student personally participates in the development of a system that offers similar features to those of many commercial packages increases the interest in the subjects and constitutes an encouragement that translates into better final grades: the methodology based on problem-solving seems quite adequate.

However, there are some learning problems which solution is too slow in the PBL environment. So, in order to increase the efficiency, Interactive Embodied Pedagogical Agents are introduced. The use of Maxine system and agents has revealed itself as a very useful instrument for guiding, informing, teaching, motivating and attracting students. The learning environment allows not only the teacher to show specific examples, but the students to easily manipulate by their own 3d scenes, experimenting the consequences of changing different parameters and options in the tutorials. This way, as stated by the students, their comprehension of complex CG concepts is improved. Moreover, the use of an agent in the learning environment opens up new possibilities based on its ability to present information to a great number of people or in a more personalized form, offering a truly multimodal interface, boosting student feelings of self-efficacy and being able to adapt itself to user needs. In spite of all what it's been said, the use of the agent presents several weaknesses: agents are currently complex to create, natural language understanding technology is in its infant stages, text-to-speech suffers from robotic voices, speech recognition technology is not strong enough for widespread use, may distract users and needs students undertaking to be useful.

To sum up, the teaching experience in Computer Graphics and the use of the agent tool has been very welcome, shows a high level of acceptance and it has ostensibly improved the opinion (and results) of students.

Acknowledgments. This work has been partly financed by the Spanish "Dirección General de Investigación" TIN2007-63025 and by the Regional Government of Aragon through the WALQA agreement 2004/04/86 and the CTPP02/2006 project.

The authors would like to thank David Anaya for his collaboration in this work.

References

- [AR07] ALTEROVITZ G., RAMONI M.: Bioinformatics and proteomics: An engineering problem solving-based approach. *IEEE Transactions on Education* 50, 1 (2007), 49–54. 3
- [BCS07] BALDASSARRI S., CEREZO E., SERON F.: An open source engine for embodied animated agents. In *Proc. Congreso Español de Informática Gráfica: CEIG'07* (2007), pp. 89–98. 4
- [CBS07] CEREZO E., BALDASSARRI S., SERON F.: Interactive agents for multimodal emotional user interaction. In *Proc. of IADIS International Conference Interfaces and Human Computer Interaction* (2007), pp. 35–42. 4
- [CHL07] COSTA L., HONKALA M., LEHTOVOURI A.: Applying the problem-based learning approach to teach elementary circuit analysis. *IEEE Transactions on Education* 50, 1 (2007), 41–48. 3
- [ERL99] ELLIOT C., RICKEL J., LESTER J.: Life-like pedagogical agents and affective computing: An exploratory synthesis. In *Artificial Intelligence Today* (1999), vol. 1600 of *Lecture Notes in Computers Science*, pp. 195–212. 4
- [GS92] GALLAGHER S., STEPIEN W.: The effects of problem-based learning on problem solving. *Gifted Child Quarterly* 36, 4 (1992), 195–200. 2
- [LCK*97] LESTER J., CONVERSE S., KAHLER S., BARLOW S., STONE S., BHOGAL R.: The persona effect: Affective impact of animated pedagogical agents. In *Proc. of CHI'97* (1997), pp. 359–366. 4
- [LCS*97] LESTER J., CONVERSE S., STONE B., KAHLER S., BARLOW S.: Animated pedagogical agents and problem-solving effectiveness: A large-scale empirical evaluation. In *Proc. 8th World Conference on Artificial Intelligence in Education* (1997), IOS Press, pp. 23–30. 4
- [LRL00] LEWIS W., RICKEL J., LESTER J.: Animated pedagogical agents: Face-to-face interaction in interactive learning environments. *International Journal of Artificial Intelligence in Education* 11 (2000), 47–78. 4
- [MGJ06] MARTI E., GIL D., JULIA C.: A pbl experience in the teaching of computer graphics. *Computer Graphics Forum* 25, 1 (2006), 95–103. 3
- [MRK06] MASSEY A., RAMESH V., KHATRI V.: Design, development, and assessment of mobile applications: The case for problem-based learning. *IEEE Transactions on Education* 49, 2 (2006), 183–192. 3
- [RN98] REEVES B., NASS C.: *The Media Equation: How People Treat Computers, Television and New Media Like Real People and Places*. CSLI Publications, New York, Stanford University, 1998. 4
- [Sav00] SAVIN-BADEN M.: *Problem-Based Learning in Higher Education - Untold Stories*. Open University Press, 2000. 2
- [vBK*04] VAN KAMPEN P., BANAHAN C., KELLY M., M'CLOUGHLIN E., O'LEARY E.: Teaching a single physics module through problem-based learning in a lecture-based curriculum. *American Journal of Physics* 72, 6 (2004), 829–834. 3