A Multimedia Authoring Tool for Language Instruction

Alexander Nakhimovsky, Tom Myers, Alice Nakhimovsky

Computer Science Department, Colgate University, Hamilton NY 13346 USA

KEYWORDS: multimedia authoring tools, language instruction, interface design

AFFILIATION: Computer Science Department, Russian Department, Colgate University

E-MAIL:	sasha@cs.colgate.edu tommy@cs.colgate.edu alice@cs.colgate.edu
FAX NUMBER:	315-824-7831
PHONE NUMBER:	315-824-7586

0. Abstract

We describe a multimedia-based Russian-language course at Colgate, and a multi-media authoring system that makes the creation of such courses technically feasible for language instructors. The system uses high-level visual tools to integrate the various media components of the educational unit and to establish links between them. Although designed with language instruction in mind, the system can be used for creating educational materials in any domain, as long as they consist of video/animation clips, accompanying texts, and glossaries or commentaries.

The system has been created using ScriptX, a multimedia authoring environment jointly developed by Apple and IBM. The environment consists of an object-oriented programming language and a class library. An earlier version of our system was based on Apple Media Kit, which consists of two parts: the Apple Media Tool, a visual environment for media integration, and the Apple Media Programming Environment, which makes it possible to extend the Tool. In addition to describing our system, we provide a comparative analysis of the two object-oriented languages and class libraries

1. Multimedia pedagogy

Multimedia – in the sense of digital video and sound integrated with the more traditional text and graphics materials – is becoming technologically feasible and affordable. It has reached the stage where it is poised to move from business executives' seminars and pilot projects into thousands of classrooms around the world. The question of what multimedia is, educationally, good for, is thus becoming urgent: multimedia development should be driven by pedagogical goals, not technological capabilities. The answer, to a great extent, depends on the educational subject matter. We concentrate on language instruction, traditionally one of the most labor-intensive fields of education. We attempt to approach the problem in all its aspects, by putting together a comprehensive development and testing environment that covers the entire range from authoring tools design to usability studies.

In the Fall of 1994, the Russian Department at Colgate taught a course whose "reading matter" consisted of excerpts from a Russian movie and its text, as well as accompanying pedagogical materials (glossary and commentary). The course was both a popular and pedagogical success. Both in the classroom and in the lab, students worked well, and their achievement in vocabulary, grammar, and intonation was unusually high. In addition, the course taught us valuable lessons in multimedia pedagogy, in user interface design, and in multimedia authoring tool design.

Pedagogically, we have established two principles, which may seem fairly obvious in retrospect, but took us some time to formulate. First, video materials for language study work well if they have a richly-textured background full of authentic local detail. There is room, of course, for a "talking heads" discussion of deep issues, for a stand-up comedian, or for an animation to illustrate a grammatical point, but the main video ought to be about people and situations that are perceived as "real" and "different". The second principle, related to the first, is that the people on screen should be totally removed from the situation of classroom and language learning, and they should talk about something that is vitally important to them. It does not matter whether the video is a documentary or a feature film as long as the subject matter of the class is not so much language learning as it is an effort to investigate and understand the workings of a different society and people in it.

Our main emphasis is thus on the content of the video materials rather than on the interactive and multidimensional capabilities of multimedia. We sought to minimize the number of buttons on the screen, providing only the navigational controls for moving around in the one-dimensional movie and for studying its text. By observing our students in the lab, we found that their actions typically followed the following pattern: watch a segment of the video, perhaps several times; read the corresponding text; invoke text support materials (dictionary, commentary) as necessary; go back to the next segment of the video. It followed that the "action language" of the interface must support those tasks in a straightforward way. Unfortunately, the authoring system that was used to create the materials for the course was far from ideal in that respect, and we set out to develop our own.

2. The authoring tool

In designing the authoring tool, we assumed that a language instructor may want to work with the following kinds of materials: movies (including animations), sound, pictures/graphics, and text. A typical configuration may include a movie clip, its text (script), a dictionary or glossary of some sort, and a commentary to the script and/or movie. Alternatively, one may want to have a text with accompanying pictures, a sound track and, again, a dictionary and a commentary. In either case, the component elements must be connected by a system of links: the user must be able to move from the movie to the corresponding place in the script and vice versa; from a word in the script to its dictionary definition or to the comment in the notes; and so on. The authoring system we are developing makes the task of putting such materials together easy for the teacher. Once the teacher has placed the digitized movie and all the requisite text or RTF files in the appropriate folders (directories), establishing the links is done without any programming, in a point-and-click fashion. For instance, to establish a link between a position in the movie and a location in the text, the teacher would stop the movie by clicking on the stop button of the QuickTime control bar, then click at the appropriate position in the text window. In addition to manually created hypermedia links, there is also a search engine and an interaction window for entering search strings. Another interaction window can be used for testing: the student is presented with questions, and the student's answers are saved to a file that the teacher can later inspect.

In more detail, as of current writing (November, 1995) the system provides for the following student activities:

A. Interactions with the movie:

- Watch the movie without interruptions
- Watch one segment of the movie and stop
- Replay the last-watched segment
- Watch the next segment
- Display the text of the current movie segment in the text window

B. Interactions with the text

- Show dictionary definition of the selected word
- Show commentaries on the selected region of text
- Play the movie corresponding to the selected region of text

C. Testing and class management

The program presents the student with review questions. The answers are saved to the student's personal log file that the teacher can inspect at any time.

In a typical lab, the student is assigned a movie excerpt to prepare. The student works through the segments of the movie, looping through a segment as many times as necessary, inspecting the corresponding text, and working with that text using the text support tools. When the student expresses readiness to take a test based on the excerpt, the system switches into the testing mode. In the current version, once the system enters the testing mode, the access to support materials is disabled, although the student can still play the movie. (The student can, of course, quit and restart the program, but that fact becomes recorded in the log file.) In future versions, this behavior will be one of several possible behaviors specified by the teacher.

The teacher actions are as follows:

A. Movie-Script coordination

The most labor-intensive part of preparing the course is establishing the links between the video and the text. This can be done in two ways. The teacher can simply play the movie, clicking at appropriate text positions to link them to the corresponding movie positions. Alternatively, the teacher can specify the characters (usually, punctuation marks) that establish tentative text segment breaks. These are registered by the system. The author can then cycle through them looking for the next segment break that he or she really wants. Once such is found, proceed as follows: play the movie, stop it at the corresponding moment, and add the text-movie position pair to the list of such pairs.

If desired, the teacher can add a symbolic name (e.g., “The First Quarrel”) to the text-movie position. This option might be useful for more advanced courses where the segments will be longer.

B. Text support

Much text support is automated. Once the teacher has placed the dictionary in the specified folder, the built-in search engine will look up words in the text that are double-clicked on. Similarly, if the teacher creates an index to the text and the commentary, the index search mechanism will find the indexing categories of selected text and the corresponding passages in commentary. However, the teacher also has the option of manually creating links between specified positions in the text and the commentary file. The procedure is similar to the text-movie synchronizaton (and uses the same user interface): the teacher selects a position in the text, clicks on the corresponding position in the commentary, and adds the pair to the list of text-commentary pairs.

3. Ongoing work and future plans

Our current development plans include extending the tool so it can create materials for languages with two-byte character sets (Chinese, Korean, Japanese). We are also working on extending the course-management component of the tool, providing the tools for record keeping and grading. Finally, we are working to provide the author/instructor with language analysis tools, such as morphological analyzer and concordance generator.

In addition to practical development, we are pursuing a program of theoretical investigation into multimedia pedagogy and user interface design to support it. Several aspects of the development environment at Colgate make it a good testbed for such an investigation. Since software developers work in close cooperation with language instructors, their designs get immediate feedback from the user. The language instructors, who are themselves designers of the end-product for the student, can design and re-design their end-products easily and fast, and test them in the classroom. This creates another fast turn-around cycle which brings usability feedback to the system developers. Finally, the system creates a framework of cooperation between the creators of video materials (documentary film makers and animators) and the courseware and software developers.

Acknowledgements

The work presented in this paper has been supported by the Colgate University Information Technology Services and by the Mellon Foundation grant to Colgate University.