Patient Simulation Using Seamless Digital Video

Gregory Cooper* and Gregory Owens**
*Section of Medical Informatics and **Division of Pulmonary and Critical Care Medicine
Department of Medicine, University of Pittsburgh, Pittsburgh, PA 15261

ABSTRACT

This manuscript describes a method for using digital-video technology to create patient simulations in which the simulated patient is always active on the computer screen. We outline the technical method we have developed, and we present the lessons learned in applying the method to develop a prototype patient simulation.

INTRODUCTION

Computer-aided instruction has been in use in medical education for over two decades. The advantages of computer-aided instruction [3] include that it provides interactive learning, individually tailored instruction, immediate student-specific feedback, safe experimentation, exposure to a wide variety of clinical situations, objective testing, and an entertaining method of learning. We focus in this paper on computer-aided instruction based on patient simulations. Most such simulations have been based on keyboard input and textual output, and many different simulation methods have been explored [3, 5, 6]. More recently, graphics and video patient simulation systems have been developed [1, 2]. Primarily, these video-based systems use videodisc technology with either keyboard and/or voice input. When a student requests information of a simulated patient, the computer finds the appropriate video clip on the laser videodisc and then plays this clip on a video monitor that is separate from the computer monitor.

A limitation of the videodisc approach is the time required to locate a clip on the videodisc — typically, about one or two seconds. While this time lag may seem short, it is sufficiently long to prevent patient simulations in which the patient is continuously present and active on the monitor. Instead, with a videodisc, the patient either freezes in a single pose, or the monitor goes blank for a second or two. In our experience such interruptions significantly detract from the realism of patient simulations. This paper presents a method for using digital video to provide continuous patient simulations. We call such simulations *seamless*, because there is no break in the activity of the simulated patient who appears on the monitor.

METHODS

In this section we discuss the simulation representation, as well as the methods used for asking questions of the patient and obtaining responses.

Knowledge Representation

A patient simulation system clearly requires a computer representation of simulated patients. The space of possible representations is large, and includes static question/answer associations, Markov state-transition models, rule-based representations, and frames. Currently our representation primarily consists of a simple hierarchical tree of question/answer associations. While this representation is straightforward, it provides an adequate approach for the initial investigation of seamless patient simulations. Each node in the hierarchical tree contains the following four pieces of information: (1) text containing the question to ask of the patient, (2) optionally, a voice-recognition profile of the question for use if speech input is used, (3) text containing the answer to the question, and (4) an integer that uniquely specifies the video clip that contains the patient answering the question.

Techniques for Asking Questions

The student user currently can ask questions of the simulated patient by pointing and clicking on a menu, by speech input, or by a combination of the two. The menus are hierarchical so that clicking on a question in a given menu will often lead to the display of another menu with follow-up questions. For example, clicking on the question "Are you short of breath?" in one menu leads to the patient answering this question and then the display of another menu containing questions such as "How severe is it?" and "How long have you had it?" The menu-based interface currently is implemented using HyperCard on the Apple Macintosh computer.

Using speech input is identical to using clicking within menus, except that speech is used to ask a question. When using speech input, the relevant menus still appear and the user at any time has the option of asking a question either with voice input or with a mouse click.

We currently are using the Voice Navigator speaker-dependent, discrete voice recognition system, which is connected to a Macintosh. The system requires that each user train it with the verbatim questions that can be asked of the system. We do not view speaker-dependent speech input as a practical question-asking approach in general, because training requires too much time. We are using it, however, to explore the degree to which speech input enhances the realism of interacting with a patient simulation. *Continuous*, speaker-*independent* voice recognition is likely to be available on personal computers within the next year, and this technology is likely to be much more practical for use in patient simulation systems.

The Technique for Answering Questions

The focus of this paper is on the method used for answering a question that a student asks a simulated patient. In this section, we present a brief overview of digital video in general and then discuss the particular way we use it to create a seamless simulated patient who responds to questions.

An Overview of digital video

Digital video is a method that stores video (and audio) on a computer hard disk and plays it back onto the computer monitor. Recently, digital video has become widely available on personal computers, such as the Macintosh and IBM-compatible personal computers (PCs). Currently available digital video systems include Digital Video Interactive (DVI) for Intel-based computers, QuickTime for the Macintosh, and Microsoft Video for the Windows system. The key technical problems that digital video has had to solve are (1) handling an enormous quantity of video information in real time, (2) providing standards to facilitate software and hardware development of applications, and (3) making the systems easy to use.

The first problem has been addressed by using compression and decompression algorithms (codecs) to reduce the amount of information that must be processed by the computer and stored on hard disk. It is not unusual to attain compression ratios of 20 to 1, which means that 100 MB of raw video information can be stored using only 5 MB of hard disk space. The decompression routines are fast enough to permit randomly accessed video clips to be played in real time from a hard disk.

The second problem has been addressed by major manufacturers establishing open standards that other vendors can follow in developing products, such as video editing software.

Solving the third problem has made digital video technology available as a high-level tool that allows users to focus on the application rather than on implementation details. QuickTime, for example, is a transparent data type, which means that a QuickTime video clip can be copied and pasted between applications, just as text can be. Also, QuickTime takes care of all timing considerations, so that, for example, the same video can be played on different type Macintosh computers (e.g., a Mac II and a Quadra 950) and still look basically the same, even though one Macintosh is a faster machine than the other. Also, video editing software, such as Adobe Premiere, is now available that supports advanced editing features that are easy to use.

Applying digital video to patient simulation

We believe that improving the realism of simulated patient cases will increase their educational use and value to medical students. We have developed a prototype of a method that produces seamless digital video (using QuickTime [4], version 1.5) of a patient on a computer screen. In particular, the simulated patient waits for the medical student to ask a question. When given a question, the patient on the screen gives a reply by looking at the student and speaking. During periods when no questions are being answered, the patient blinks, looks around, and in general behaves like someone who is waiting for the student clinician to ask or do something. The video simulation is seamless because there are no breaks in the video between answering questions. That is, the patient on the screen never disappears, freezes, or jerks, but rather, always seems to be continuously and naturally present. To achieve seamless simulation, we developed a computer program that dynamically splices together a set of video clips of the patient in order to generate a continuous video image of the patient. In the remainder of this section, we explain the main ideas underlying our approach.

At any given moment during a patient simulation, one of two types of video clips is playing on the computer monitor. One type of clip contains the patient answering a student user's question. We call these the *answer clips*. The other type of clip shows the patient waiting for another question. We call these the *wait clips*. During wait clips the patient may be blinking, coughing, fidgeting, or doing any activity that a real patient might do while waiting for a clinician to ask another question. We currently use 45 wait clips that last about one to two seconds each.

Our primary technical problem was how to enable the computer to dynamically splice together answer clips and waits clips so that the entire video appears seamless. We accomplish this task by having each answer and wait clip begin and end with the patient in a standard position. Thus, the end of any answer clip is identical (or almost identical) to the beginning of any wait clip and vice versa. When the final frame of one video clip contains the same image as the first frame of another video clip, we say the clips *match*. Since we do not know how long the simulated patient may need to wait before answering a question, we may have to splice together multiple wait clips. As long as all the wait clips match, the waiting period will appear seamless, regardless of how long it may last.

Figure 1 schematically illustrates the two types of clips. Here we show two possible answer clips (with spoken text at the bottom) and three possible wait clips. Figure 2 shows a dynamic video that is created by a specific interaction with a student user, which we now explain. When the simulation begins, the simulated patient appears in wait state A, then in wait state B. During wait state B (i.e., during clip 2), suppose the user asks for the reasons the patient has come into the clinic. Clip 3 in Figure 2 shows that Answer A from Figure 1 is dynamically spliced into the video after clip 2 finishes playing, and by playing clip 3 the simulated patient answers the student's question. After clip 3 finishes playing, we have the simulated patient return to waiting by having wait state C play, which is shown as clip 4 in Figure 2. Suppose that during the playing of clip 4 the student asks how long the patient has had a cough. Clip 5 in Figure 2 shows that answer B from Figure 1 is dynamically spliced after clip 4. This cycle of question, answer, and wait continue until the student says goodbye to the patient.

There is one important elaboration to the procedure just described. We sometimes allow the simulated patient to volunteer information or make unsolicited statements. For example, if the student user takes too long in asking a question, the patient will say, "Excuse me, but are you going to ask any more questions?" If the student still does not ask a question within a few seconds, then the patient angrily says "I'm sorry, I have to go now. Bye.", and then leaves, at which point the simulation stops.

A PROTOTYPE APPLICATION

We have developed a prototype of the seamless digital-video method for a patient who has acute bronchitis. The ultimate purpose of this simulation is to have the student recognize that the patient smokes, and therefore, the student needs to take appropriate steps to try to help the patient stop smoking. Such a simulation will make it cost effective and logistically feasible to have many medical students learn how to handle this clinical situation appropriately. Eventually, we plan to expand the suite of simulations to include a wide variety of patients and clinical problems.

The interface

The user interface for the HyperCard implementation of the simulation system is shown in Figure 3. The menu shows some of the symptom questions that currently can be asked by the student through clicking on the question or using speech input. Although not shown in Figure 3, some menus contain category items (e.g., "Ask the patient about symptoms"), which branch to menus that contain questions about that category. Typically, after the simulated patient answers a question, the menu changes to display additional follow-up questions that the student may ask. The arrow at the top of the display allows the user to return to the previous menu. The window on the right side of the interface shown in Figure 3 continuously displays a motion video (with sound) of the simulated patient. The quality of the video image appears poor in Figure 3 because it is a black and white version of the color image that appears on the computer monitor.

Technical issues

In the this section we discuss technical issues we encountered in creating this patient simulation. We used VideoSpigot hardware to capture the patient video directly from an 8mm Sony camcorder, although the capture could have been done from videotape. The video was captured at 12 frames per second (fps), because we discovered that this is the maximum rate the Macintosh IIci can handle and still keep the sound synchronized with the video. The filming was done all at one sitting by using a script that contained all possible wait states and all answers to allowable questions. The person being filmed tried to assume a neutral face and a standard head position at the beginning and end of each wait clip and each answer clip being filmed.

The total number of answers was 28, including 3 different ways to answer "no" and 5 different ways to answer "yes." The questions appeared across a total of 9 menus. The total duration of all 28 answer clips is 194 seconds, and the total duration of all 45 wait clips is 54 seconds. The total set of answer and wait clips required 14 MB of hard disk storage when using the VideoSpigot codec for data compression.

LESSONS LEARNED

We are encouraged by the level of realism we have been able to achieve so far with the initial prototype. In developing this prototype we have learned some important lessons that we hope will facilitate and enhance the future development of digital video patient simulations by our group and others. In this section, we summarize these lessons.

Using a video rate of 12 fps is minimally adequate for capturing realistic clips of a talking patient. Occasionally, however, at 12 fps a moderately rapid movement of the patient will make the video look jerky. We believe that a rate of 20 fps will be sufficient to eliminate this problem. The higher end Macintoshes are able to attain a 20 fps rate with images that are about 4 times the size of the one we used. Defragmenting the hard disk before filming is important, since the throughput of a disk is optimized when it is defragmented, and such optimization is important for the data rates required for digital video. If the disk cannot keep up with the video, then QuickTime is forced to drop some video frames, which leads to a jerky image on playback.

The mechanics of filming, while easily overlooked at first, are very important. We have found that the camcorder autofocus should be off, and the use of proper lighting is essential to obtain a good image of the patient and avoid distracting shadows. To avoid capturing the noise from the computer and hard disk, a directional microphone is helpful. The video clips are much more likely to closely match if an entire simulation is filmed at one time, because it is easier for the patient-actor to maintain a standard head position and facial expression so that the clips will match. The chair in which the patient-actor sits should be stable and fixed relative to a static background scene. We also provide a hidden head rest to help the patient-actor maintain a standard head position.

To increase the chance that most of the video clips will match well, we film clips that contain small amounts of patient movement before clips that contain relatively more movement.

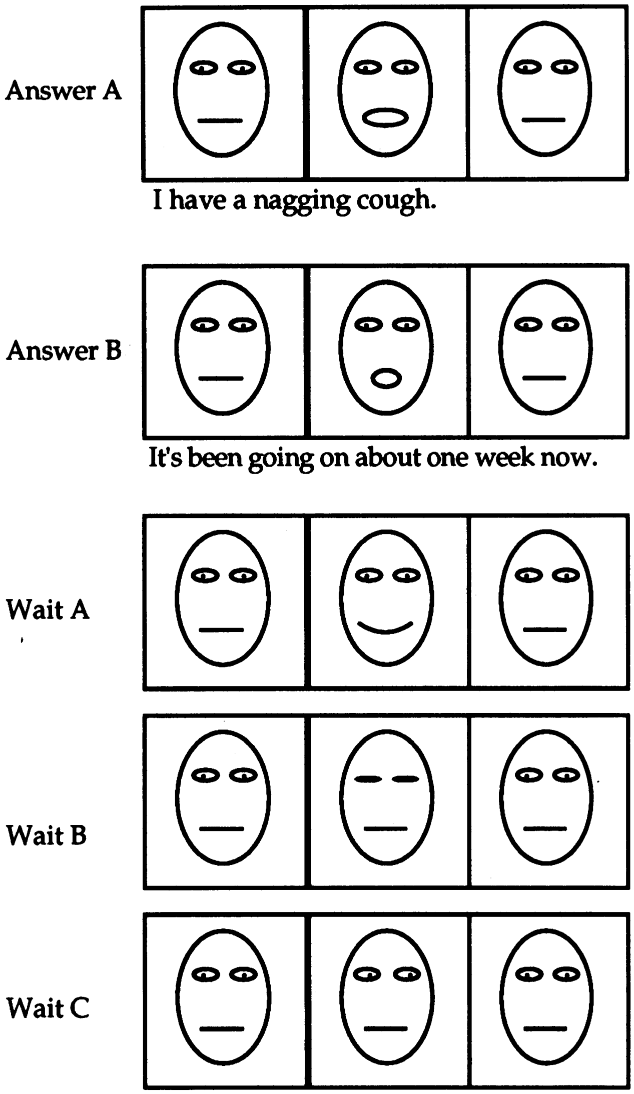

Figure 1. Each three-frame object represents a set of video frames that form a video clip. The images in this figure are shown as schematics for illustration only. The actual video clips contain the images and voice of a real person. The top two clips show a patient answering questions. The bottom three clips show the patient waiting for a question to be asked. Note that the first and last frames are the same for all five video clips, and thus, the clips all match each other.

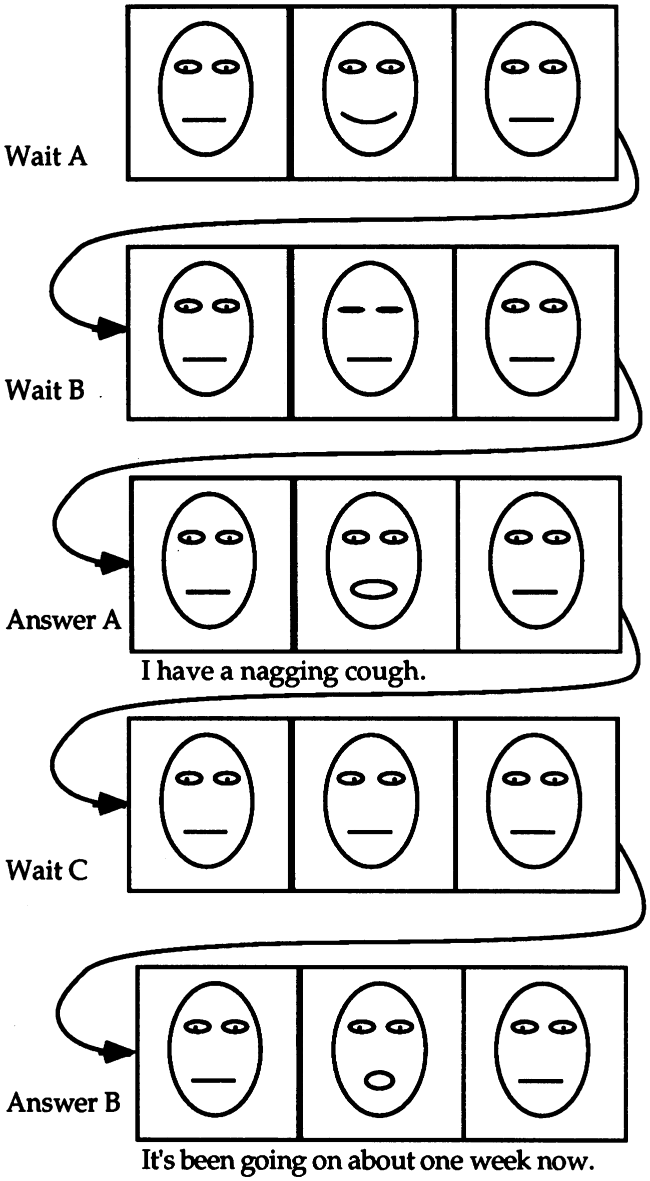

Figure 2. Here we show the real-time computer-based construction of a patient simulation based on the dynamic splicing of video clips from Figure 1 in response to questions asked by the student user. An actual patient simulation contains many more clips than are shown here.

Since it is virtually impossible to get perfect matches between all clips in a simulation, we are investigating the use of computer-based methods that automatically transition in a smooth way from the end of one clip to the beginning of another. In particular, we are using a technique called *morphing*, which produces spatially warped crossfades between images. With morphing, one image (in our case, the last frame of some clip *x*) appears to gradually become another image (in our case, the first frame of another clip *y*). More specifically, we are applying Gryphon Software's Morph program to morph the first and last frame of each clip to a single frame of a neutral face. Thus far, the resulting clips match almost perfectly. If an initial match is good (yet not perfect), simple crossfades appear to be as seamless as more complex spatially warped crossfades.

Based on our current experience, we believe that with a well organized production facility a programmer/editor and an patient-actor can create a single patient simulation containing about an hour of answer clips in about one day, *if* a script is available, which contains all answers that must be filmed.

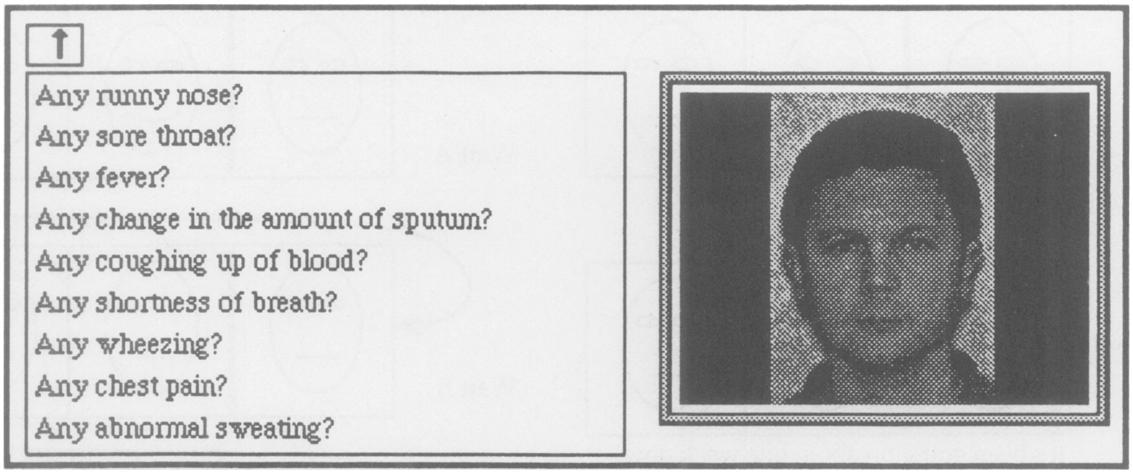

Figure 3. The patient simulation interface. The menu on the left is used to ask questions by clicking with a mouse pointer. Clicking the arrow displays the previous menu of questions. The window on the right contains the video image of the patient (actual size) who is being interviewed. The image on the computer monitor is in color and is of much higher quality than the image shown here, which was generated with dithering by a black and white laser printer.

We note, however, that deriving a script of the appropriate questions and answers is likely to require substantial amounts of time before filming can begin.

We have found that any changes in mood while answering a question should gradually transition back to the neutral face at the end of the answer clip, otherwise the patient looks artificial. Currently, the postage-stamp size patient image is smaller than we would like. We believe an image that is approximately 3 by 3 inches would be a significant improvement, and attaining this size image is feasible with current high end personal computer hardware.

DISCUSSION

The seamless digital video method we introduced in this paper can be used with a wide variety of patient simulation models that are more sophisticated than the one we have presented. Seamless simulation is a general method that has the potential for wide applicability. Even more elaborate methods are likely to be possible in the future. In particular, we believe that software will become available for computer-controlled, dynamic editing of digital video clips. This will allow more sophisticated simulations, including dynamic fades between different patient scenes (e.g., a fade from a patient being in the clinic to being in a hospital room), and the automated zooming in and out on the simulated patient, as appropriate in the context of the interview.

We have several near term goals for improving the simulation. We plan to film each answer clip multiple times with the patient actor portraying a different mood (e.g., neutral, happy, melancholic, etc.) when giving an answer. Mood-specific wait clips also will be filmed. The simulated patient can transition from one mood to another by way of wait clips that make the mood transition. Other near term goals include increasing the video image size, expanding the set of questions that students can ask, and observing medical students beta test the system before we design a formal evaluation to investigate the educational value of the seamless patient simulation method.

Acknowledgements

We thank the following people for their generous help: Desi Butter assisted in early programming efforts. Henry Lowe provided helpful discussions about multimedia, HyperCard, and QuickTime. Greg Gross permitted us to use his multimedia computer system to develop an initial prototype. Constantin Aliferis allowed us to use his 8mm camcorder. Constantin Aliferis and Henry Lowe provided helpful comments on an earlier version of this paper. This research was supported by NIH grant NHLBI-5-K07HL02104.

REFERENCES

1. Abdulla, A., Watkins, L. and Henke, J., The use of natural language entry and laser videodisk technology in CAI, *Journal of Medical Education* **59** (1984) 739.
2. Harless, W.G., Zier, M.A. and Duncan, R.C., A voice-activated, interactive videodisc case study for use in the medical school classroom, *Journal of Medical Education* **61** (1986) 913.
3. Hoffer, E.P. and Barnett, G.O., Computers in medical education. In: Shortliffe E.H., Perreault L.E., Wiederhold G. and Fagan L.M. (eds.), *Medical Informatics* (Addison-Wesley, Reading, MA, 1990) 535.
4. Ito, R., QuickTime in action, *MacUser* May (1992) 84.
5. Piemme, T.E., Computer-assisted learning and evaluation in medicine, *Journal of the American Medical Association* **260** (1988) 367.
6. Starkweather, J.A., The computer as a tool for learning, *Western Journal of Medicine* **145** (1986) 864.