Merging Lecture Video with Slides

Background

Many instructors post their lecture slides to course web sites. Video recording lectures is less frequent, but is becoming more practical with the availability of inexpensive video cameras that record directly to flash memory, such as the Mino series and its competitors. Better than posting a video of the lecture, though, would be posting the sound track of the video with the images (mostly or entirely) replaced by lecture slides.

Example Scenario

Professor Lupin lectures partly from slides and partly using the blackboard. He asks a student volunteer to record the lecture with an inexpensive flash-based video recorder. After the lecture, he uploads the video to his computer. He also exports his PowerPoint slides as JPEG images, in the same directory with the video, and opens up the JPEGs and the lecture video in *Lectern*, an application created by University of Oregon students. *Lectern* displays a series of video frames in several rows. For a 50 minute lecture, 5 rows of 10 video frames are enough to show frames at 1 minute intervals, which is enought to determine roughly where each slide transition takes place.

He drags images of slides to positions in the array of video frames. It takes a couple minutes to align them roughly with slide transitions in the lecture. Next he clicks one of the placed slides, bringing up a more detailed row of video frames around that time, and uses a slider to indicate more precisely when display of the slide should begin and end. He does the same for each placed slide, sometimes letting one slide transition directly to the next, and sometimes leaving a bit of "talking head" video between them. He leaves longer sections of video showing portions of the lecture on the blackboard, and demonstrations of zombie combat techniques.

Seven minutes later, he has produced a simple video of his lecture in which the sound track is entirely from the video recording, but most of the video has been replaced by images of his slides. It isn't professional quality, but it is good enough, and he posts it to his course web site for students.

Basic Requirements

The basic requirements of this project are to make it easy and fast to produce *good enough* videos of lectures, combining recorded audio from the actual lecture but replacing some or all of the video images by higher quality images of slides. The emphasis is on *fast* and *easy*, rather than professional quality. If an instructor cannot produce the merged presentation in about ten minutes, he or she is very unlikely to take the trouble to produce it (despite likely having spent hours preparing the lecture in the first place).

Lectures are fairly dynamic ... instructors may skip forward in a set of slides, skip some slides, back up to reiterate a point, answer a question that doesn't directly relate to any of the slides, and illustrate a point on the blackboard. A video preparation system should accommodate these variations, although it should leverage the common case of sequential advance through a set of slides to speed the video presentation process.

Room for Creativity

While it will be worthwhile to study the interfaces of other video editing applications, like iMovie and Adobe Premier, there is an opportunity here to devise user interactions that are more specialized to the task at hand. Generate a lot of ideas before winnowing and refining them.

Since the available time is short for a project like this, another opportunity for creativity (really, I'm serious) is finding a good way to cut it down to a subset or prototype that is feasible in the time you have. Scoping a project and breaking it into manageable pieces is a key software engineering skill, and not at all an easy one.

Main Challenges

Unless you are already familiar with video file formats and APIs for manipulating them, a main challenge will be quickly learning enough to design your approach. You will probably spend a lot more time reading API documentation than actually writing code.

A second challenge, mentioned above as an opportunity for creativity, is to scope the project appropriately. You must produce something of value in four weeks. You probably can't produce a complete, polished application in that time. What can you produce? Perhaps you can produce a useful component of the complete application. Perhaps you can produce a bare-bones version that provides a few critical features. Perhaps you can produce a quick and dirty prototype that is adequate for evaluating different approaches to the user interface. Whatever it is, you should think very carefully about what the value of your product is.