Screencasting Facility for Software Demonstrations

Abstract

This report outlines some experiments in recording software demonstrations for use in teaching. Two methods of screen capture are compared.

Keywords

screencasting, software demonstration
Introduction

On the Audio and Music Technology (AMT) pathway, it is not unusual for teaching staff to arrive at a lecture and spend considerable time setting up specialist audio hardware and software in order to demonstrate a piece of software. On occasion, the hardware and/or software does not function correctly at first, and it proves necessary to return to the AMT studios for additional equipment or technical support. The aim was to set up a facility allowing staff teaching on this pathway to record screencasts of audio software demonstrations. The intended outcomes were that lectures could be delivered more effectively, both in class and online.

Methods

A variety of software and hardware options were considered. These are summarised in table 1.

<table>
<thead>
<tr>
<th>Capture Method</th>
<th>Hardware</th>
<th>Software</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Local software</td>
<td>Computer</td>
<td>Echo360 Personal Capture</td>
<td>Already available to staff</td>
<td>Reduced processing power</td>
</tr>
<tr>
<td>(2) Remote software</td>
<td>Computer, Second Computer, Frame grabber</td>
<td>Any camera encoder</td>
<td>No loss of processing power</td>
<td>Extra hardware</td>
</tr>
<tr>
<td>(3) External Hardware</td>
<td>Computer, scan converter, DVD recorder</td>
<td>None</td>
<td>No loss of processing power</td>
<td>Extra hardware</td>
</tr>
</tbody>
</table>

Table 1: Possible methods of screen capture

Method 1 (Local software) uses a program installed on the lecturer’s computer. Several packages are available. Kadirire (2011) discusses these, and compares Echo360 and Panopto. Since the former is available to Anglia Ruskin staff, this is the most obvious one to try. While this package can be used to record the lecture using audio visual (AV) equipment, the current project was primarily focused on just recording the screen, so the Echo360 Personal Capture variant was used (Everett, 2011). It should be noted that this can record from a webcam.

It was hypothesised that this method would be adequate when running relatively low-power applications such as PowerPoint, it would not be as successful with more demanding applications.

Method 2 (Remote software) removes all computational demands from the presenter’s computer, and instead carries out the capture on a second computer. The key device in this is known as a frame grabber. This splits the monitor (HDD-15) signal from the presenter’s computer into two; one copy goes to the monitor or AV system; the other is converted into the signal produced by a digital camera. The second computer connects to this via USB, and records this using any software compatible with DirectShow (e.g. Microsoft Media Encoder, VideoLAN VLC or RealProducer).

This would permit the capture of more demanding programs, and the output of the frame grabber is not compressed. However, the resultant video still requires compression, and thus the resultant video quality depends not only on the resolution and refresh rate of the first computer but the speed of the second computer. It was hypothesised that a combination of these factors would limit the usefulness of this method.

Method 3 (External hardware) uses a scan converter (a device which converts the monitor signal to S-video). This could then be recorded by a standalone DVD recorder. Again there is no restriction on the complexity of program being recorded. The frame rate should be high, but unfortunately the relatively low resolution of S-video is likely to be the limiting factor in this method.

It was decided to test the first two methods.

Experimentation

While the original intention had been to focus on audio applications, much of the experimentation for method 1 used a piece of electronics simulation software called Electronics Workbench. This is shown below.
As well as the screen, the Echo360 system can record inputs via a microphone and webcam. In this case it was decided not to use a webcam. An inexpensive clip-on microphone is sufficient, but considerable trial and error is needed in setting the correct microphone volume. For each recording the first step is to plan and rehearse the presentation. In some cases the circuit was constructed ‘live’; in others a pre-built circuit was used. When a significant mistake is made the options are to start again or to spend time editing the video – in practice the first option was chosen as the videos are relatively short and for the first run there was a higher emphasis on completing the demonstration than on making it flawless.

When the file is ready the easiest procedure is to find it (not straightforward), then both copy it onto a memory stick and upload it to http://myplayer.anglia.ac.uk. It can be played from MyPlayer in a lecture if there are appropriate AV facilities, but it is generally quicker to play it from the memory stick. Often a video introduced in one week was replayed the following week in order to review the topic. Videos were also linked directly from the VLE page for the module.

The experiment on method 2 was not connected to a particular module, but was used for a talk given by two colleagues to the Audio Engineering Society. This was done to evaluate the frame grabber approach, and also because the presenters’ Macintosh laptops did not have Echo360 installed. The talk was also filmed on two video cameras; the audio was recorded, and the output from the presenters’ laptops was recorded. The presentations used a combination of PowerPoint, Prezi (another presentation tool), and Logic Audio. The hardware used for this was an Epiphan VGA2USB LR frame grabber (Epiphan, 2012). In this case the output from a Mac was recorded on a PC using the Epiphan software capture tool. Several AVI codecs were tested; the Indeo 5.10 codec (recommended by Epiphan) was found to be at least as good as any other. The three videos were to be edited together separately.

**Evaluation**

First, some comments on Echo360. It achieves its main objectives. The files are quite heavily compressed, resulting in some loss of quality, and this issue is currently being examined by Anglia Learning and Teaching. This was something of a surprise given that much of the source material is monochrome and thus in principle amenable to a high compression ratio. Also, the interface could be improved so as to allow files to be renamed and stored in a user-specified location. The editing facilities in Echo360 were not tested. A total of 12 videos were made using this technique.

The MyPlayer server was also found to largely achieve its goals. One limitation found (the inability to delete a file), was promptly fixed. However the metadata cannot be edited, and the full titles of each video are not shown on the index page. Linking to MyPlayer from the VLE revealed a bug in the latter’s hypertext editor, again since rectified.

Use in class showed the screencasts to be effective. Since the software is not available on the university network, it allowed the demonstrations to be embedded into the lecture rather than the practical sessions,
and eliminated the possibility of time being wasted due to the software not working correctly. Perhaps more importantly, it was clear that students were also accessing the material in their own time, taking advantage of the ability to stop/start/replay a presentation. Each video was accessed an average of 29 times by the 41 students on the Digital Electronics module. If multiple views could be discounted, this would correspond to 71% of the class; however it is not possible to discount this possibility.

The second experiment with the frame grabber was a partial success; no processing power was taken from the presenters’ computers, and setup time was minimal. However, the Mac resolution had to be reduced from 1440×900 to 1280×800 (other frame grabbers permit higher resolutions.) Although the frame grabber’s output is uncompressed, the need for real-time compression on the recording computer limited the output file to around 5 frames per second, watchable but jerky.

It is not possible to directly compare the two approaches, as they were applied to different machines running different programs at different resolutions. However it is fair to observe that both suffered from restrictions in the resultant video quality.

Conclusions and Future Work

In general, the outcomes were largely achieved. The first approach, Echo360, is appropriate when the software being demonstrated is not too processor-intensive. Importantly, the students were accessing the materials in their own time as well as seeing it during the lecture. A further outcome is that the videos can be used in subsequent module deliveries.

The second approach, using a separate recording computer, addressed the case of processor-intensive software. This approach met its more limited goal, but the low frame rate suggests the need for further optimisation.

Future work is likely to concentrate on quantifying the trade-offs between the computational resources and the resultant video quality, and on determining how and when students are using the materials generated.

As a final recommendation, Echo360 is appropriate for recording programs of low computational complexity at low frame rates. Those wishing to record the use of more complex software should consider the use of a frame grabber and a high-powered second computer.

References

