Chapter 8

Video Storage and Transmission

This chapter introduces the architecture and realization of E-Chalk's video subsystem that belongs to the project almost since its inception. It discusses the general purpose of video transmission for electronic chalkboard lectures before a more enhanced method is presented in the subsequent chapter.

8.1 Preliminary Considerations

When instructors do not want to adapt to a new technology or educational institutions are not able to invest in electronic chalkboards, lectures held with a blackboard can still be captured by recording a video of the lecturer acting in front of the board. As discussed in Chapter 2, several educators use standard Internet video-broadcasting systems for the transmission of all kinds of lectures. The primary advantage of recording a lecture with a camera is that the approach is rather straightforward. Well-known techniques can be used for recording, and off-the-shelf Internet broadcasting software can be used for digitizing, encoding, transmission, and playback. The lecturer's work flow is not disturbed, he or she does not have to get used to a new teaching medium. Even though some projects have tried to automate the process [Gleicher and Masanz, 2000, Rui et al., 2001], recording a lecture the "conservative way" requires additional manpower for camera and audio-devices operation. Yet the video compression techniques used by traditional video codecs are not suitable for chalkboard lectures for the same reasons as discussed in Section 5.5: Video codecs mostly assume that higher-frequency features of images are less relevant, which produces either an unreadable blurring of the board handwriting or a bad compression rate. In addition to artifacts, non-electronic chalkboard drawings are sometimes also difficult to read because of low contrast. Figure 8.1 shows an example of a traditional chalkboard lecture compressed with a typical video codec.[1]

Using an electronic chalkboard (see Chapter 3) is a better alternative: It captures strokes and allows to save them in a vector-based format. Vector-based

[1] For further reading: A very different approach that provides a partial workaround for the problem presented here can be found in [Wallick et al., 2005].

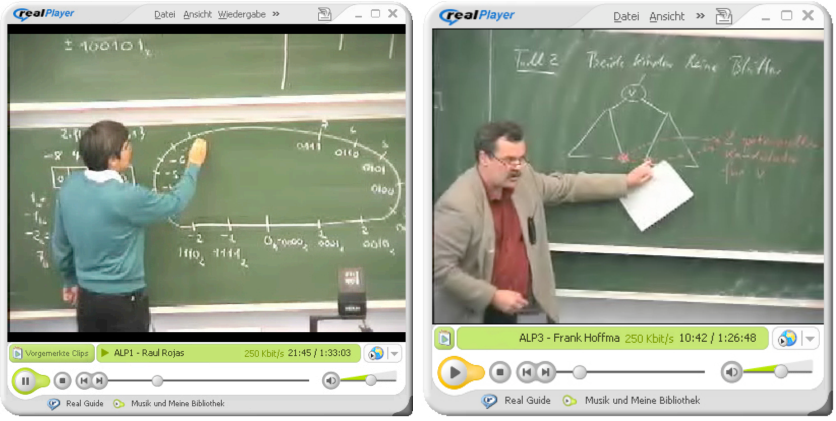

realPlayer
Datei Ansicht Wiedergabe »
Vorgemerkte Clips
ALP1 - Raul Rojas 250 Kbit/s 21:45 / 1:33:03
Real Guide
Musik und Meine Bibliothek
realPlayer
Datei Ansicht »
ALP3 - Frank Hoffma 250 Kbit/s 10:42 / 1:26:48
Real Guide
Musik und Meine Bibliothek

Figure 8.1: Two chalkboard lectures captured and played back with commercial Internet broadcasting systems. Due to lossy compression and low contrast, the chalkboard content is difficult to read. The lectures where given and recorded at Freie Universität Berlin.

information requires less bandwidth, can be transmitted without loss of semantics, and is easily rendered as a crisp image on a remote computer (compare Chapter 5). Still, a disadvantage that was reported to us by many students is that during distance replay the objects on the board appear out of nowhere. The lecture appears impersonal because there is no one acting in front of the board. The replay lacks important information because the so-called "chalk and talk" lecture actually consists of more than the content of the board and the voice of the instructor. Often, the facial expression of the lecturer adds to the verbal communication or the instructor uses gestures to point to certain facts drawn on the board. Sometimes it is also interesting to get an impression of the classroom or lecture hall. Psychology suggests (see for example [Krauss et al., 1995]) that gestures and facial expressions are part of a person's semantic of encoding ideas. The understanding of words partly depends on gestures as they are also used to interpret and disambiguate the spoken word [Riseborough, 1981]. All these shortcomings aggravate with the creation of board content being temporarily abandonded for pure spoken phases or even non-verbal communication. In order to transport this additional information to a remote computer, the E-Chalk system provides an additional video server. As shown in Figures 5.3 and 8.2, the video pops up as a small additional window during lecture replay. The importance of the additional video is also supported by the fact that several other lecture-recording systems (compare Chapter 2) have also implemented this feature, and the use of an additional instructor or classroom video is also widely discussed in empirical studies. Not only does an additional video provide nonverbal information as to the confidence of the speaker at certain critical points, like irony [Dufour et al., 2005]. Several experimental studies (for an overview refer to [Kelly and Goldsmith, 2004]) have also provided evidence that showing the lecturer's gestures has a positive effect on learning. For example, [Fey, 2002] has reported that students are better motivated when watching lecture recordings with slides and video in contrast to watching a replay that only contains slides and audio. [Glowalla, 2004] also shows in a comparative study that students usually prefer lecture recordings with video images over those without.

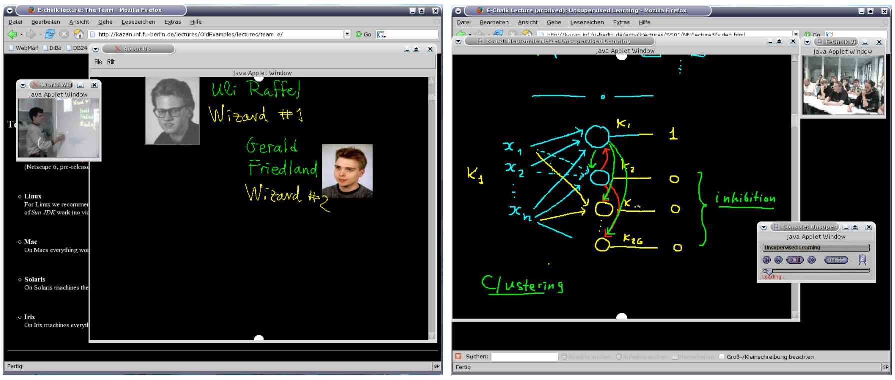


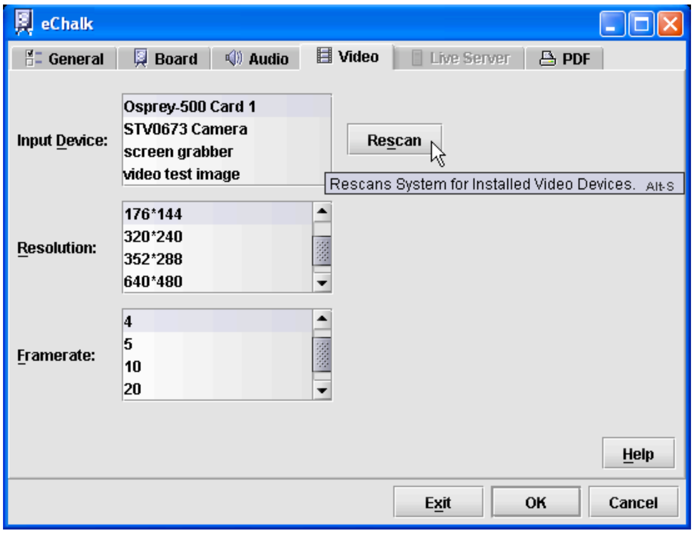
Figure 8.2: Two examples of the use of an additional video client to convey an impression of the classroom context to the remote viewer.

Yet, this transmission of non-verbal information requires several additional resources. A camera is needed for capturing, handling the video stream consumes CPU time on both ends, and the additional video data requires a huge amount of additional storage capacity and bandwidth for transmission. The E-Chalk video system is therefore an optional component. The classrom as well as the student side can choose to turn it off. The video system compresses the video data down to a manageable size and deals economically with memory and CPU resources on both sides.

8.2 Overview

In the beginning of the E-Chalk project, the development of the video subsystem was guided by the same idea as the World Wide Radio 2 audio system (see Chapter 6). Initially, the video system was called *World Wide Video (WWV)* and aimed at building "a fully featured Internet video streaming system that runs on any hardware or platform" [Friedland et al., 2002]. During the beginning of the development of the video subsystem, the system did not only inherit the idea of WWR2, it also inherited its problems. The processing of video data is usually more expensive since there is more information that has to be handled. For this reason, the system was built as an asymmetric system based on the assumption that the server side has rather unlimited resources while the client provides only low computational performance. Since E-Chalk has mainly been built to support one-to-many communication, this paradigm still governs the architectural approach of the system.

The E-Chalk video system is mainly divided into three parts. A system configuration and hardware detection part, the actual server, and the receiving client. The system configuration and hardware detection directly interacts with the E-Chalk Startup Wizard. The server is divided into a set of SOPA nodes that allow grabbing, processing, encoding, transmitting, and converting video content. The Java-based replay client is described in Section 5.2.3. The video server is similar to the audio server described in Chapter 6. The video server consists of a set of SOPA nodes. In the default configuration without instructor segmentation, only four nodes are used: a video capturing node that delegates



Input Device: Osprey-500 Card 1, STV0673 Camera, screen grabber, video test image

Resolution: 176*144, 320*240, 352*288, 640*480

Framerate: 4, 5, 10, 20

Figure 8.3: The video configuration panel of the E-Chalk Startup Wizard.

to JMF, an encoder node that compresses the incoming frames, a node that writes the compressed data to a file for archived replay, and optionally a server node that streams the encoded data live to any connecting client.

8.3 Configuring the Video Server

As described in Chapter 4, the Jave Media Framework provides a platform-independent means to access video hardware. E-Chalk uses the Java Media Framework for capturing and auto-detecting video devices. Auto detection is encapsulated in a generic media node. It detects a wide range of hardware devices on different platforms. In addition to grabbing from hardware plugged to the computer, the E-Chalk video system also provides two virtual capturing devices for testing purposes: a screen-grabber records consecutive screen-shots and a test pattern device creates a video from a user-defined set of test images. Figure 8.3 shows the GUI panel of the video configuration. The user can choose an input grabber, the frame rate, and the video resolution. The settings defined in the E-Chalk Startup Wizard are written to property files that are read by the SOPA Framework.

8.4 Video Encoding

The video codec in E-Chalk is a very simple, lossy motion compensation codec. The designing goals of the codec prioritize simplicity and computational efficiency rather than trying to achieve a maximum compression ratio. Appendix F shows a detailed syntax description of the format.

The encoder creates three types of frames, called *I-Frames*, *T-Frames*, and *0-Frames*. I-Frames are *JPEG images*[2]. I-Frames can be used to improve the

[2] Strictly speaking, the term “JPEG format” is incorrect. The right term is JFIF (JPEG

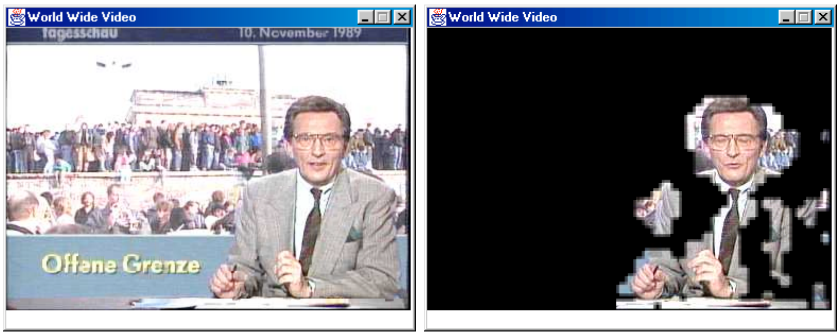


Figure 8.4: Visualization of the encoding of two consecutive frames from a TV broadcast (source: ARD, "Tagesschau", November 10th, 1989): Blocks that have not changed significantly from the left frame to the right frame are colored black.

quality of the video replay, for example at the beginning of a new scene, at the cost of bandwidth. The first frame recorded or sent when a client connects is an I-Frame.

T-Frames (transparency frames) are generated by a simple motion compensation mechanism. The redundancy of static parts of a scene over several frames is utilized. T-Frames consist only of those blocks in the picture that have changed significantly or have aged. All other blocks are tagged as transparent and are encoded as zeros. The player substitutes uncoded blocks with older ones from previous frames. A block, as in JPEG, is an 8 × 8-pixel matrix. The difference is calculated in the YUV color space with the components weighted Y:U:V as 4:1:1[3] The pixelwise Euclidean distances between the current frame and the previous frame are summed up for each block. If the sum of the pixel changes in a block is n times bigger than the average sum of all blocks, the block is defined to have *changed significantly*. Figure 8.4 demonstrates the approach. If a block has never changed significantly over a period of t seconds, it is defined to *have aged*. Because of this ageing strategy, the video is self-repairing and I-Frames are not required. The variables n and t are set to $n = 4$ and $t = 2$ by default, but may be adjusted by the user in order to control the degree of compression.

For random seek, the video is played back beginning from the specified T-Frame. This individual T-Frame does not contain all image blocks. This results in several parts of the image containing black 8 × 8-pixel holes. However, after two seconds of playback, all blocks must have been filled because the ageing rule forces an update of each block after two seconds. Encoding a 640 × 480-pixel video is easily possible in real time. On a Pentium 4 3 GHz the algorithm encodes such a video file with more than 50 frames per second. In practice, much lower frame rates are used. This leaves enough CPU resources for the other tasks running during lecture recording and transmission.

0-Frames contain no image data at all. They are used as a placeholder to skip one frame and are transmitted when not a single block is marked opaque.

file interchange format), whereas the specification can be found in [ISO/IEC JTC1, 1994]. However, the term "JPEG format" is used more commonly.

[3] The reason for this is that the human eye is more sensitive to contrast changes than to color changes. This is a standard heuristic used in several image and video encoding standards.

0-Frames are encoded using a single byte and thus need less bandwidth than a T-Frame with all blocks marked transparent.

The incoming images are consecutively encoded. A sequence of 20 frames forms a packet. Each packet is then compressed using the GZIP format [P. Deutsch, 1996]. The compression ratio obtained is roughly 40:1. In the E-Chalk system, mainly a quarter picture of NTSC (that is 192 × 144 pixels) and four frames per second, is used to obtain a bandwidth of about 64 kbit/s. Experience shows that this frame rate is acceptable when the video is transmitted in a separate window and only used as an additional source of information. The actual bandwidth required for a certain transmission, however, finally depends on the video content.

This encoding strategy is used for both the regular video encoding and the overlaid instructor video (compare Chapter 5). In the latter case, the client interpretes the color black as transparent. The next chapter will explain the idea behind the instructor extraction approach.