Integration of a Video Annotation Tool into a coactive Learning and Working Environment

Johannes Magenheim[1], Wolfgang Reinhardt[1], Alexander Roth[2]
Matthias Moi[1], and Dieter Engbring[1]

[1] University of Paderborn, Department of Computer Science, Computer Science Education Group, Fuerstenallee 11, 33102 Paderborn, Germany
{jsm,wolle,moisun,didier}@uni-paderborn.de
[2] University of Paderborn, Faculty of Business Administration and Economics, DS&OR-Lab, Warburger Strasse 100, 33098 Paderborn, Germany
roth@uni-paderborn.de

Abstract. In this paper we describe pedagogic scenarios where the use of a video annotation tool could be of added value to the students' overall learning process. Furthermore, we introduce ViLM, a platform-independent tool for annotating videos collaboratively or on its own and how we integrated the tool in our university's LMS. Finally, we characterise potential research opportunities and learning scenarios where the tool may successfully be applied.

Keywords: Computer Science Education, Training, Video, Learning Management Systems

1 Introduction

The Computer Science Education Group at the University of Paderborn started the usage of multimedia and hypermedia in their student teacher training program in 1998. The aim of these efforts was to support student teachers to integrate new technologies into their teaching and learning processes and use multimedia as a very convenient tool for self-assessment of their educational practice in the classroom (see Magenheim and Schubert [1,2]). At the core of these practices is the video recording of real teaching situations and the annotation and discussion of teaching and learning strategies. The most important lessons learnt for the participating student teachers are specific strategies of pedagogies of computer science as well as the analysis and evaluation of social classroom interaction and the design and implementation of excellent lessons in computer science.

Ever since many tools were developed that aim at the individual competence development and lasting learning successes. One of these tools is ViLM (Visualization of Learning and Teaching Strategies with Multimedia) that underwent multiple upgrades; re-implementations that used advanced programming languages for the best fit to the organizational requirements. Despite these iteration cycles there were still a number inadequacies of the tool: the missing integration into the university's LMS, the separate storage of video data and additional course materials, missing streaming

video capabilities, unavailable access to university user data stores, leading to the absence of global authentication procedures.

Based on these issues we started to reengineer the existing source code and designed a new version of the ViLM tool that makes use of our university's IT and digital learning infrastructure. The new VILM tool supports a variety of video annotation scenarios with differing permission structures and cooperative functionalities.

This paper introduces several pedagogic application scenarios for the usage of a video annotation tool, the coactive learning and working environment *koaLA* used at the University of Paderborn and the newly developed tools ViLM ControlCenter and ViLM Player. Furthermore we describe detailed application scenarios for the tools mentioned above and we discuss future research opportunities.

2   Pedagogic application scenarios for video annotation in educational contexts

There are many application scenarios for the use of video annotation in educational contexts using a different number of videos. In the following, selected scenarios are introduced that can make use of the meaningful and supporting application of video annotation tools in general and ViLM in particular.

2.1 Talks and Presentations (one camera perspective)

Each talk deals with a more or less bounded topic and it is of high importance for the audience being able to follow the talk. Therefore a sensible structuring and a golden thread are of high importance. There are certain things that can be analysed regarding a presentation including a) is the diction appropriate? b) are various media used during the talk? and c) are those media used meaningful? or d) how is the chain of reasoning constructed? and e) are mimic and gestures supporting the told? Those aspects are a selection of criteria that can be taken into account during the evaluation of presentations. Without using ViLM an observer could jot down notes during the talk or use a recording of the talk afterwards. The presentation itself and the according notes would in each case be separated from each other resulting in media disruptions [3].

In order to enhance the traceability for the presenter ViLM can be used. Therefore a recording of the talk is necessary. The camera should be oriented towards the speaker and follow his movements if applicable. For doing the analysis, the talk can be split into phases (e.g. according to the outline) with remarks to every aspect of interest during the phase. There can be different aspects of interest in different phases and it is easy to switch between them using ViLM. The speaker himself, an educational guardian or a collaborative group of students, can do the analysis of a talk.

2.2 Film analysis (one camera perspective)

For doing film analysis no actual recording but instead a digital copy of the film is

necessary. During a film analysis several criteria can be assessed including a) art work through the camera, b) art work after the recording, c) composition of stylistic elements, d) dubbing and e) décor of the film. During a written film analysis there is no direct connection between the notes and the described situation. Normally one would write down the time signature of a specific scene but there is no marker on the video nor can a specific time be headed to.

During a film analysis with ViLM different phases and events can be defined and directly marked in the video. A detailed analysis (by adding a description or linking to additional material) can be done for each of these annotations single-handedly or cooperatively. The markers set on the video can directly be started up making the navigation through a film analysis much easier.

2.3 Analysis of teaching strategies (two camera perspective)

According to Blömeke et al. [3] lessons can be divided in designated phases that can be used exemplarily during the analysis of teaching strategies: a) problem, b) target agreement and significance, c) procedure agreement, d) development of foundations for problem solution, e) problem solution, f) result comparison and summary, g) application, and h) continuation and rating. Of course every student teacher is free to structure his lessons on his own.

The application of ViLM in teacher training fosters student teacher's awareness of their own behaviour in the classroom, especially regarding reactions on unforeseen situations – during group discussion at the teacher training course it can be debated whether they were appropriate. The analysis of teaching strategies involves two camera perspectives: a camera that records the (re-)actions of the teacher and another camera that records the (re-)actions of the pupils. Having two simultaneous perspectives on the same situation facilitates the direct comparison between action and reaction. Thus, discussions among students about behaviour patterns of student teachers and pupils may be activated.

2.4 Analysis of group work (four camera perspective)

Learning co-operatively in teams is a very common method in the classroom where the number and size of groups depends on the number of pupils. In this scenario we assume four groups with one camera oriented on each of them and the simultaneous start of all four recordings. The task of the group work needs to be explained in beforehand and in order to achieve the objectives, all group members have to cooperate. To wrap up the teamwork and to present the different results a final result presentation is suggested. The recordings of the group work allow analysing a) which pupils have actively worked in their group, b) who made decisions and c) how the problem solution was worked out. ViLM facilitates the definition of different phases and the annotation of actions performed in the teams. The parallel screening of all four camera makes it easy to compare problem solution approaches and activities as well as the identification of opinion leader and passive observers. ViLM enables the teacher to gain a better insight into the single group works and a better understanding of the

general scheduling of the group work task. From this analysis he or she then can optimise the group assignment, time management, and work description for future realisations. Furthermore, the teacher can show different approaches for the problem solution to his pupils by showing the recordings to them.

In this paper we present the integration of video annotation software into the coactive learning management environment *koaLA* that is used organisation-wide at the University of Paderborn. The specific functionalities that support coactive learning in groups are introduced in the following section.

3 The coactive learning and working environment *koaLA*

CSCW/L systems got great impulses from the discussion about web 2.0 or e-learning 2.0. Beside the use of already mature technologies like RSS, web services and Asynchronous JavaScript and XML (AJAX) for implementing open environments and user-friendly interfaces, the variety of options for cooperative working and learning has steadily increased. In combination with social networks, tools like wikis, weblogs and podcasts emphasise the learner's role as content producer respectively its cooperative activities. This development leads away from completely predetermined structures and closed contexts to knowledge- and individual-centric, open environments.

In the course of a support programme of the Federal Ministry of Education and Research, the project *Locomotion* (Low Cost Multimedia Production and Organization) was executed at the University of Paderborn. This project constituted the implementation scope of the virtual learning management environment *koaLA* (Coactive learning and working, http://koala.upb.de), which focuses on individuals and individual cooperation contexts and combines this focus with conventional features of traditional course management systems. Important elements and necessities of learning, of university administration and of social networking are integrated through virtual and – depending on the context – self-organised learning and working areas. Here, the consequent alignment of individual requirements of learners is fundamentally. The implementation of *koaLA* as an open system with its focus on individually configurable learning and working contexts is well accepted by students, but also by teachers using the virtual environment to try out innovative didactic concepts and integrating a potpourri of pre-structured and self-structured scenarios into different types of courses.

In the following, we want to explain the main concepts and the functionalities, which enable the configuration and seamless combination of formal as well as informal contexts. After that, we describe the technical implementation by means of typical web 2.0 architectures and demonstrate, how differences in media formats in the university's information architecture were neutralised through integration on different levels.

The coactive learning and working environment *koaLA* was set up as an application on top of a standard architecture for web-based CSCW/L systems named *sTeam*[1], which provides the fundamental functionality for cooperative learning and working through programming interfaces. In its core, the server combines two simple but ele-

---

[1] Structuring Information in a Team, http://www.open-steam.org

mentary concepts: On the one hand, generalization of media types, and on the other hand, the utilization of virtual knowledge spaces for implementing varieties in group structures and working contexts. In addition to a flexible rights management, the generalization of media types is an elementary precondition for a platform-wide inter-linking of information, without struggling with differences in media formats. Fur-thermore, virtual knowledge spaces describe the semantic structuring of a knowledge area and bring together various services and communication mechanisms at a single location. The aim here is to reduce media discontinuities as introduced by Keil [4], when using different media functions for manipulating material. For instance, knowl-edge objects (any kind of media type) can provide the basis for a discussion (e.g. be annotated), or user groups can correspond to mail groups.
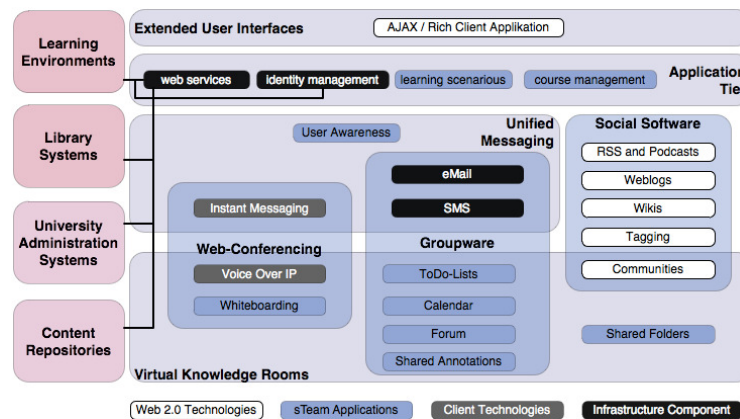

Fig. 1: The *koaLA* architecture

For the purpose of proclaiming the concept as a predominant structural characteris-tic for such a heterogeneous environment as a universities it-infrastructure, the server applies some common internet-protocols onto the elements of the metaphor of virtual knowledge spaces. Therefore, synchronous as well as asynchronous communication tools like instant messaging, whiteboarding, e-mail or shared annotations are natively embedded in knowledge rooms.

In the application tier, dynamic learning and working contexts are reconstructed by means of these functional components (cf. Hampel and Roth [5]). From a technical perspective, those depend often on the same functionality, but – in the end – vary in the degree of freedom in self-organization. Also, the integration of some of the uni-versities central basic and complex services is carried out in the application tier.

The presentation layer of *koaLA* allows a unitary and easy handling in combination with a visually attractive web interface by means of AJAX-functions. This should lower user barriers and increase the users´ acceptance. As an example, the sequence of lectures can be rearranged through the drag-and-drop of a single lecture, without the necessity to reload the web page.

In the past, the *sTeam* server already provided basic services for different learning and community environments. However, new aspects in *koaLA* are a distinctive focus on social networking functionalities and the embedment of modern cooperative tools

as weblogs and podcasts next to wikis and forums.

With its mass of configurable and dynamic options, different learning scenarios can be realised easily without programming skills, only through the combination and adaption of reusable functional components. As examples, the variety of scenarios includes the editing and exploitation of documents, the structuring of discourses, the rating and arrangement of media elements, and the coordination of activities, which are scattered in time and space. In particular, learners can organise themselves in small groups; they can easily jump from their own workbench to a private group to a cooperative workroom of a course and way back. This concept offers great possibilities for learners to switch between informal and formal learning contexts and collect, exchange or publish documents and all kind of media objects - without any direct dependency on a special course of lectures.

4　Technical Aspects of ViLM

In this section we will discuss the technical aspects of ViLM and give a brief overview about the technologies used, the design decisions taken and the overall architecture of the ViLM project. Furthermore we introduce the new ViLM ControlCenter and the ViLM player.

4.1　Requirements Engineering

Based on the decision to integrate the new ViLM tool into the coactive learning and working environment *koaLA* and the already existing legacy applications and code, we had to analyse the application domain for the new ViLM tool and to derivate functional and non-functional requirements for the tool. Since *koaLA* is a web-application that can run in any browser with basic JavaScript support, one requirement was to develop the new tools platform independently. Another design decision was to develop a Rich Internet Application (RIA) that offers the user a rich and engaging user experience and comes with the characteristics of desktop applications. During the design phase we evaluated the Flex[2] framework by Adobe and the JavaFX[3] framework and decided to use Flex because of its mature status and the excellent support for video display and video streaming. Furthermore, Adobe Flex guarantees high portability between the common operating systems Mac OS X, Windows and Linux. Flex not only allows developing web-based RIAs but also desktop equivalents using the AIR runtime. With the AIR runtime additional functionality like full access to the local file system and drag and drop of files into the AIR application is realised. In another project we developed an Adobe AIR based video player that allows the playback of annotated video materials and the presentation on additionally stored material. During the requirements engineering phase we decided to reuse the existing code base of the player and to evolve it to be the ViLM player, capable of streaming videos and interacting with *koaLA*.

---

[2] http://www.adobe.com/products/flex/
[3] http://javafx.com/

During the design phase we decided to use streaming video instead of video play-back via HTTP. The usage of a streaming server allows a more precise navigation and a smoother display of even large video files. For annotating videos and synchronisation of multiple views it is of high importance to navigate between key-frames of video files as precise as possible. To avoid costly re-encoding of video material in order to suit the capabilities of a streaming server, we were looking for a streaming server that was capable of dealing with H.264 encoded MP4 videos, streaming via RTMPS (Real Time Messaging Protocol over HTTPS) and easy connection to Flash-based applications, such as any Flex or AIR application. After an evaluation phase we decided for the open source Java streaming server Red5[4] because of its very good video streaming results, the easy configuration and the RTMPS support. Another influencing variable for Red5 was the possibility of extending the existing codebases to suit special needs and adaptations to fit the existing IT infrastructure. The proprietary equivalent product from Adobe (Flash Media Server[5]) was not chosen because of its pricing structure and the missing open development community that Red5 offers.

4.2 Architecture of the ViLM project

The ViLM ControlCenter (VCC) was developed as the main interaction interface in the overall architecture. Both students and staff are using the VCC to interact with video annotation projects. The backend for the VCC is the *koaLA* learning and working environment of the University of Paderborn. All communication between the VCC and the server is event-based and uses an HTTP ActionScript API. For security reasons, all communication is secured with SSL. Using the API it is possible to authenticate and authorise against the university LDAP and to create and modify objects in the *koaLA* system. *koaLA* implements a multi-level rights management and thus supports multiple right scenarios for video annotations as well.

Whereas all project-related documents and control files are stored in a special folder of the course on *koaLA*, the respective videos are stored on the streaming server. Therefore we extended the Red5 server to handle uploads from the VCC and to authenticate any video request via the *koaLA* authentication interface. That way videos can only be viewed by legitimised individuals.

The integration of video annotations into *koaLA* was realised by the implementation of a new learning scenario (unit) that can be selected by staff members for their course in the LMS. Creating a new video annotation unit from within *koaLA* enables the VCC to create new projects within this unit. For example, a staff member could create new ViLM projects for each of the students in his course or for specific pedagogic scenarios (cf. section 2). The main control file that connects the different data sources and subsystems is an ANN-file (ANN stands as abbreviation of annotation). The file is a custom XML file), which is stored for each ViLM project within *koaLA*. The AAN-file holds information about the videos to stream, the documents used within the project and most important about marked phases and annotations in the project. The ViLM player is capable opening these ANN-files, loading the related

---

documents and streaming the video content.

4.3 The ViLM ControlCenter

The newly developed ViLM ControlCenter (VCC) is the main application for dealing with video annotations within the University of Paderborn IT infrastructure. The VCC is an AIR application and runs on the major operating systems. The VCC enables the easy creation and administration of new or existing ViLM projects and is also the main working area for students doing their video annotations. Staff members can select from a list of different video annotation scenarios and decide on various right settings for the created projects. For newly created projects staff members can set the student owner of the project and directly upload the video recordings to the streaming server. Doing this the initial ANN-file is created and stored on *koaLA*.

When students login to the VCC, they can select video annotation projects that they have access to. Depending on the applied video annotation scenario and right settings they can annotate the available videos on their own or collaboratively. The first step in the annotation process is to synchronise the available videos, so that all recordings start at the same time. During the recording we use an audible sign to identify the synchronisation point. After the videos are synchronised, students can define different phases of interaction in the videos. Those phases can correlate to the lesson planning or simply contain noticeable events. Each phase is defined by a start- and end-time and a meaningful title. Each phase can contain multiple sub-phases for further structuring of the video. Students can add any type of additional media (such as lessons planning, intended blackboard figures, used material to prepare the lesson…) to the phases. Apart from phases, students can also define events in the videos, which are single frames within the recordings that show important scenes (a resulting blackboard figure or a special kind of interaction). The annotated phases and events are saved in the ANN-file and the references to the added documents are linked and everything is uploaded to the project store on *koaLA*.

Apart from creating new (only staff members) and annotating existing projects (mainly students), all users can view completely annotated video annotation projects using the ViLM player. The player can directly be launched from within the VCC.

4.4 The ViLM player

The ViLM player is responsible for the streaming of videos from the streaming server and the display of the annotated phases, events and uploaded documents. The ViLM player comes with the well-known controls (play, stop, pause) of a video player and has an additional phase menu, where all defined phases can directly be accessed. The ViLM player displays descriptions and comments to the current phase and allows direct access to the linked documents. The ViLM player uses the ANN-file to stream or download the respective files and to display the descriptions and comments. With the player it is possible to display selected phases and videos in full screen mode and to control the volume of the available videos. Doing this it becomes feasible to concentrate on a specific view of the project.
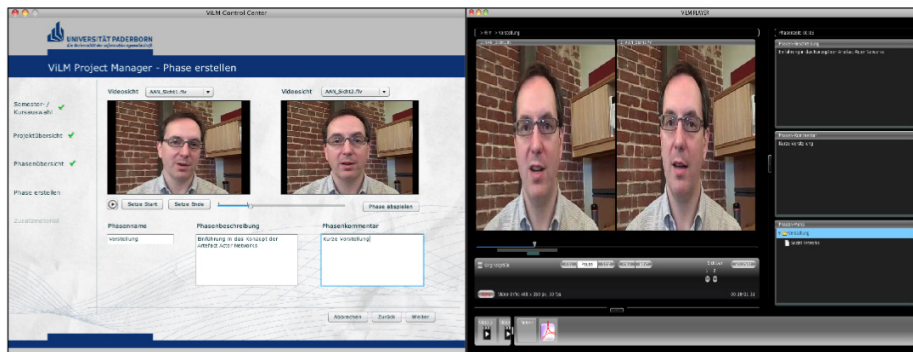
Fig. 2: Screenshots of a project in the ViLM ControlCenter (left) and the ViLM player (right)

5 Detailed Inspection of selected application scenarios

In this section we take a closer look on two selected application scenarios where the ViLM software is applied.

5.1 Self-reflecting teaching strategies in teacher education

In a problem-based course with the topic 'methods of teaching informatics in theory and practice' (MIU) at the University of Paderborn, student teachers should become acquainted with typical classroom situations. Therefore the course takes place at several locations. One location is the University of Paderborn where two staff members (tutors) supervise the student teachers. Furthermore up to three schools in Paderborn are involved where the student teachers observe lessons on computer science and perform own lessons. The teachers in school act as mentors for the student teachers. The MIU course is divided into three phases:

1. Learning to prepare lessons in computer science,
2. Performing lessons in computer science and learning to reflect on them,
3. Reflecting again and generalise best practices.

Phase 1: Learning to prepare lessons in computer science

In the first phase of the course the main goal is that student teachers become acquainted with typical classroom situations and learn about theoretical assumptions about teaching computer science. They should be prepared to develop their own course materials and to plan computer science lessons together with their fellow students. Materials and concepts of teaching will be delivered in form of web-based documents on the university coactive learning and working environment *koaLA* and can be accessed by all participants of the course.

During this phase of the course student teachers also visit schools to attend lessons

in informatics and to analyse the teaching strategies of teachers and the pupils' concepts of learning. Theoretical assumptions about teaching informatics and practical experience of classroom situations will be discussed with their tutors in the course at university.

Phase 2: Performing lessons in computer science and learning to reflect on them

After having been prepared for school practice the student teachers have to teach one lesson of computer science in the classroom on their own. They will be observed by some of their fellow students and the mentors using a jointly developed observation form. The communication and collaboration in the classroom is recorded by two video cameras, one for the teacher's and one for the students' perspective (two camera perspective).

In school, the student teacher and the mentor discuss the lesson after it is over. So the teacher students may have a better chance to analyse and to reflect on their own lesson. After his own lesson the student teacher uses ViLM to watch the recordings of his teaching and annotates demonstrative phases of the videos, describes them and adds additional material if applicable. At university, one student teacher presents some annotated clips of his own lesson with the ViLM player to the other students, which have not necessarily seen this lesson and to the tutors. Referencing the real performance with the planned schedule combines theory and practice.

Another discussion of the lesson follows to learn about different possible teaching methods in the shown situations. Doing this, evaluation remains a social process, supported by video, and the result of inter-subjective social negotiations of meanings through interpersonal discourse. On the other hand, evaluation becomes a process of individual reconstruction, supported by video. The fleeting event and experiences of own teaching becomes a solid artefact, which can be used as an external memory for all teacher students involved in this course. All materials and concepts of teaching – including the annotated video – will be delivered in form of web-based documents on *koaLA* and can be accessed by all participants of the course.

Phase 3: Reflecting again and generalise practices

Based on theories of constructivism, the course wants to facilitate the development of critical evaluation of learning and teaching processes by letting the students analyse their behaviour in the classroom by themselves. To foster this concept the teacher students continue to use the ViLM tool in order to reflect and annotate their own teaching strategies. Within ViLM they prepare a multimedia document containing all relevant materials from their own lesson (e.g. the video recordings, lesson designs, work sheets, results of processes, pictures of the blackboard etc.) and the respective annotations and descriptions of their work. The main objective of this process of reconstruction of classroom reality by student teachers is to establish sensibility towards effective teaching strategies and classroom interaction. In a final wrap up of the MIU course all student teachers present their final ViLM project to their fellow students and the tutors of the course.

5.2 Computer Science Tutor Training

At the computer science department of the University of Paderborn there is a student tutor training course in place where all new student tutors are prepared for their practice. According to Reinhardt and Magenheim [6], a core goal of this course is to prepare future student tutors for the needed shift in perspective from being a student to being a teacher and the different operational levels they will act in. To accomplish these goals the student tutor training is a two-part course with one part before the start of the semester and one part during semester. The first part consists of an online phase, where the future tutors learn to work with the administrative and cooperative functions of *koaLA* and an introductory workshop. Amongst others, various situations from tutorials are simulated during that workshop, recorded and discussed with the participants in order to reveal common errors, prepare for unforeseen situations, train the usage of multiple media and to allay existing fears. The recordings are viewed directly after the simulation so the usage of ViLM would not be appropriate here.

During the semester several meetings take place with the course members and common problems are discussed. Each student tutor is assigned to an observation group and gets observed by the other members two times each semester during his tutorials. Additionally, the observers have to fill out a detailed observation form where they can state special findings and hints for the student tutor. The observed tutorial gets recorded (one camera perspective) and the resulting video is provided to the tutor afterwards. Therefore, a mentor creates a new ViLM project and defines the owner of the project. For privacy reasons only the tutor himself and the mentors of the course can access the project directly. The other members of the course only see the phases of the video that are shown by the tutor in the workshops using the ViLM player.

Together with the observation form and the recording of the tutorial a student tutor then can replay his own teaching activities and annotate noticeable phases. In a mid-semester and an end-semester workshop all student tutors present noticeable phases from their tutorials using the ViLM player and link them to the remarks on the observation forms. Furthermore, the mentors define specific topics of interest (e.g. usage of media, behaviour during group work, reaction on questions from the audience) that should be presented explicitly. All presented video phases are discussed together with the tutor, respective observation group and the other members of the tutor training.

6 Conclusion and future R&D perspectives

In this paper we introduced ViLM, a platform-independent tool for collaborative video-annotation and its integration in the coactive web 2.0 LMS of our University, called *koaLA*. We described the advantages and the requirements of the visualised analysis of learning and teaching strategies with multimedia in different learning scenarios. Based on the requirements analysis of those scenarios, the main components of ViLM, the ViLM ControlCenter (VCC), and the ViLM player were developed. VCC enables the easy creation and administration of new or existing ViLM projects. VCC is the front-end for the students' video annotations and it is integrated

in the coactive learning environment *koaLA*. The ViLM player provides students with simultaneous video streams according to the requirements of the related learning scenarios.

Up to now, mainly the analysis of two types of learning scenarios in the area of CSE were successfully put into practice, each of them supported by video streams which can be annotated by the students. Further developments of the tools will focus on the realisation of additional scenarios with multiple video-streams: movie analysis; teamwork analysis; discussion analysis, e.g. between developers and customers in the area of computer science requirements engineering. Based on our experiences and those of other projects dealing with video analysis of social interaction scenarios [7,8] the technical development of VILM should be continued: Integration of a scene graph editor in VILM, that allows students to arrange the scenes in a graph that represents intersections and a non-linear flow according to content related criteria; assembling of a video-snippet-collection (VILM-Tube) that contains relevant learning and teaching scenarios; integration of an annotation tool, that enables social tagging of snippets according to a 'learning process ontology' (LPO) and by this means facilitates information retrieval. Finally, empirical research on students' benefit from VILM is necessary in order to improve their teaching competences and to provide the developers with substantial information for further system enhancement.

References

1. Magenheim, J.: ViLM: Visualization of Learning and Teaching Strategies with Multimedia in Teacher Education. In: Proceedings of ED-MEDIA 1999, pp. 1593-1594. 1999.
2. Magenheim, J., Schubert, S.: Evaluation of teacher education in informatics. In: Benzie, D., Passey, D. (Eds.), Proceedings of WCC 2000, pp. 181-184, 2000.
3. Blömeke, S. et al.: Gestaltung von Unterricht – Eine Einführung in die Didaktik. 2004.
4. Keil, R.: Medienqualitäten beim eLearning: Vom Transport zur Transformation von Wissen. In: Bibliothek, 31(1), pp. 41-50, 2007.
5. Hampel, T., Roth, A.: Rapid Development of Non-Monolithic CSCL-Applications - About the Benefits of Using a Prescribed Terminology in Web Programming. In: Proceedings of the E-Learn 2005, pp. 2095-2102, 2005.
6. Reinhardt, W., Magenheim, J.: Modulares Konzept für die Tutorenschulung in der universitären Informatikausbildung. In: Schwill, A. (Editor), Commentarii informaticae didacticae, Vol 1, 2008.
7. Meixner, B. et al.: SIVA Producer – A Modular Authoring System for Interactive Videos. In: Proceedings of I-KNOW and I-SEMANTICS 2009, pp. 215-225, 2009.
8. Stephan, A. et al.: Autorentool für interaktive Videos im E-Learning. In: M. H. Breitner et al. (Eds), E-Learning 2010, Springer, Berlin, Heidelberg, pp. 143-154, 2010.