Workflow-Based Architecture for Collaborative Video Annotation

Cristian Hofmann[1], Nina Hollender[2], and Dieter W. Fellner[1]

[1] Interactive Graphics Systems Group, Technische Universität Darmstadt, Fraunhoferstr. 5, 64283 Darmstadt, Germany
[2] Center for Development and Research in Higher Education, Technische Universität Darmstadt, Hochschulstr. 1, 64289 Darmstadt, Germany
{cristian.hofmann, d.fellner}@gris.informatik.tu-darmstadt.de, hollender@hda.tu-darmstadt.de

Abstract. In video annotation research, the support of the video annotation workflow has been taken little into account, especially concerning collaborative use cases. Previous research projects focus each on a different essential part of the whole annotation process. We present a reference architecture model which is based on identified phases of the video annotation workflow. In a first step, the underlying annotation workflow is exemplified with respect to its single phases, tasks, and loops. Secondly, the system architecture is going to be exemplified with respect to its elements, their internal procedures, as well as the interaction between these elements. The goals of this paper are to provide the reader with a basic understanding of the specific characteristics and requirements of collaborative video annotation processes, and to define a reference framework for the design of video annotation systems that include a workflow management system.

Keywords: Video Annotation, Video Analysis, Computer-Supported Collaborative Work.

1 Introduction

A group of students use a web-based video annotation tool to analyse video sequences taken from TV panel discussions with regard to the use of a range of specific argumentation tactics. Their task is to mark and categorize objects and sequences within the video, to annotate these selections with descriptions and own interpretations, and to compare and discuss their results with others, also by exploring databases of already analyzed videos. Users that work with a specific video analysis software are often confronted with a large number of available tools and, consequently, with a hardly comprehensible user interface. In order to ensure a fluent course of activities, a system is required that provides information about the sequence in which these tasks are to be accomplished, as well as which tools can be used referring to a certain task. Furthermore, the application should support transitions between successive work steps.

A.A. Ozok and P. Zaphiris (Eds.): Online Communities, LNCS 5621, pp. 33–42, 2009.

Research activities in the area of computer-supported video annotation have increased during the last years. Corresponding solutions have been implemented in various application areas, e.g. interactive audiovisual presentations in e-Commerce and edutainment or technical documentations [4], [25]. In our research work, we focus on the support of collaborative video analysis in learning settings performed by applying video annotation software. A growing number of application scenarios for (collaborative) video analysis in education can be identified. Pea and colleagues report on a university course of a film science department, in which two different movie versions of the play „Henry V" are analysed by a group of students with respect to the text transposition by different actors and directors [23]. Other examples for the application of video analysis in education are motion analyses in sports and physical education, or the acquisition of soft skills such as presentation or argumentation techniques [16], [23], [25]. A large number of different research fields and approaches have been involved in Video Annotation and Analysis Research. Nevertheless, one relevant aspect has been taken little into account: The support of the analysis workflow, which comprises the management of annotation data with related tasks and system services. Thus, a majority of today's applications do not consider the needs of the users regarding a complete workflow in video annotation [12]. This is especially the case for collaborative settings. By workflow-support, we mean the facilitation of loops and transitions between the single workflow steps and tasks on the one hand. On the other hand, appropriate tools and information can be provided at the proper time, depending on the current state of the work. Consequently, we expect a reduction of the learners' and tutors' load with regard to the use of such applications and hence enhancement of efficiency.

The main contribution of this paper is the presentation of a reference architecture which is based on identified phases and tasks of the video annotation workflow. In section 3, the underlying annotation workflow is going to be illustrated considering its single phases and recursive loops that can be especially associated with the collaborative processes taking place. Our investigations addressed the specific needs of users who work in teams with a special focus on educational settings. The results are based on interviews and discussions conducted with experts and users regarding the sequence of tasks and work steps within the annotation process, as well as on a summary and reflection of the existing literature. In addition to that, we performed an analysis of the functionalities, the user interface, and interaction design of fifteen video annotation and analysis applications. In section 4, the system architecture is going to be exemplified with respect to its single elements, the interaction between these objects, as well as internal procedures within the elements of the architecture. The goals of this paper are to provide the reader with a basic understanding of the specific characteristics and requirements of collaborative video annotation processes, and to provide a structural framework for the design of video annotation systems that include a workflow management framework.

2 Related Work

Bertino, Trombetta, and Montesi present a framework and a modular architecture for interactive video consisting of various information systems. The coordination of these components is realized by identifying inter-task dependencies with interactive rules,

based on a workflow management system [4]. The Digital Video Album (DVA) system is an integration of different subsystems that refer to specific aspects of video processing and retrieval. In particular, the workflow for semiautomatic indexing and annotation is focused [31]. Pea and Hoffert illustrate a basic idea of the video research workflow in the learning sciences [22]. In contrast to our research work, the projects mentioned above do not or only to some degree consider the process for collaborative use cases. The reconsideration of such communicative and collaborative aspects requires modifications and enhancements of the existing approaches and concepts.

3 Collaborative Video Annotation Workflow

The reference architecture for collaborative video annotation relies on a already presented model that implies the single phases, tasks, and loops within the video annotation workflow [15]. We identified the phases of *configuration*, *segmentation*, *annotation*, *exploration*, and *externalization*. In the following, the particular items of these steps are going to be pictured.

Before starting an annotation project, the environment has to be configured. Participants are assigned to accounts and user groups that are associated with specific roles and access rights. Furthermore, the annotation tasks can be distributed among the individual users [18], [30]. Specific project preferences can be adjusted and the graphical user interface may be customized [7]. In video analysis projects, category systems need to be fed into the system. The also may be modified during the annotation process [5]. The segmentation, annotation, and exploration tasks can be seen as one unit. Thus, video annotators alternately segment, annotate, and need to browse own results or data belonging to other annotators or annotation projects [5], [16], [22]. This process is accompanied by data reviews, comparisons, and consequently modifications [22], [24], [28]. Annotators start chunking the video into segments they want to refer to, drawing on different apporaches [2], [3], [10], [14], [17], [22], [27]. Video segments can be defined either by a single person or by an assigned group in a collaborative manner. Thus, annotations that serve as communication contributions are resources for the coordination of collaborative segmentation activities. In some of the identified use cases, the segmentation task is partitioned and assigned to users or groups. Users continue with the annotation of these subsets and with arranging annotations into a certain order. One type of annotation is the linking of metadata or descriptive data [1]. Users may also describe observed behaviours, events, or objects within the video. In most cases, they are allowed to enter free textual annotations. In fact, other types of media formats like images or sounds are possible [10]. During the annotation phase, a further task can be the transcription of verbal and non-verbal communication [20]. In video analysis, the annotation phase also includes interpreting, rating, and reflecting. These activities can be performed either qualitatively, e.g. in discussions, or quantitatively, by means of statistic methods provided by specialized software [12], [22]. Like the segmentation task, annotation may be partitioned and distributed among different groups. When a collaborative group works separately, members need to discuss their results with other participants [6], [7], [21]. Thus, discussion is a central element within the collaborative annotation process. It is a means of agreement and consistency of different annotators' results and leads to a return to previous steps of the workflow [26]. Pea and Hoffert assume that exploring one's own

data is required to properly conduct an analysis [22]. Especially in collaborative annotation, users also need to search for results of co-annotators, experts, or other sources [16]. In addition to discussion, this can be regarded as key activity concerning revision of own results and re-entries to previous workflow steps. The externalization phase includes activities without the use of the annotation application, and consists of any kind of publishing operations. It begins with editing and converting the data into several formats, and moves on to presenting this information in corresponding media [22]. E.g., databases of annotated video material can serve as digital resource for information retrieval in subsequent annotation sessions. In video analysis, it is often necessary to export data for further inspection with specific applications [12], [22].

4 Workflow-Based Reference Architecture

In this section, we present an architecture model which is based on the workflow model pictured above. This architecture provides a basic structure for video annotation systems which include a workflow management framework. Design decisions particularly base on our endeavour to support given operating procedures within the annotation workflow. In that context, we identified basic requirements that have to be fulfilled by the architecture: *Workflow Control*: Transitions between workflow phases and the control of sequences of sub operations have to be supported. Also loops and re-entries to other phases of the workflow must be considered. *Enclosure*: The identified phases and tasks need to be pooled into functional units that are mutually delimited. Thus, task areas can be typecasted and invoked by addressing respective modules. *Extensibility*: The architecture must enable administrators to integrate, replace and remove tools that can be assigned to task-related modules. *Consistency*: Since multiple tools read and possibly write on the same data, the consistency of shared parts of the data set has to be ensured at every point of the annotation process.

In addition to that, there are further requirements with regard to annotated data and collaborative activities. Thus, an appropriate handling of media files and its annotated information, as well as their organizational structure must be provided. With respect to collaborative use cases, the architecture model has to realize the data exchange between multiple spatial separated users of the application. For this purpose, stored information must be made available to every participant of the group. Consequently, consistency of data must also be warranted for every peer in the shared system.

As showed in Figure 1, the reference architecture was conceptualized as component-based client-server model. The elements of the architecture are structured by a combination of the Model View Controller and Mediator patterns [8], [11], [29]. In the following sections, the single aspects are going to be illustrated.

4.1 Client-Server Model

A fundamental condition for collaborative processes is the interconnectedness of every peer taking part for information exchanging purposes [7], [10]. A range of optional models can be considered, e.g., client-server, peer-to-peer, or web-based approaches. We suggest a client-server architecture, not only due to its wide spreading in the area of information systems [7]. The server application realizes the centralization of the information space and, at the same time, makes the data system available

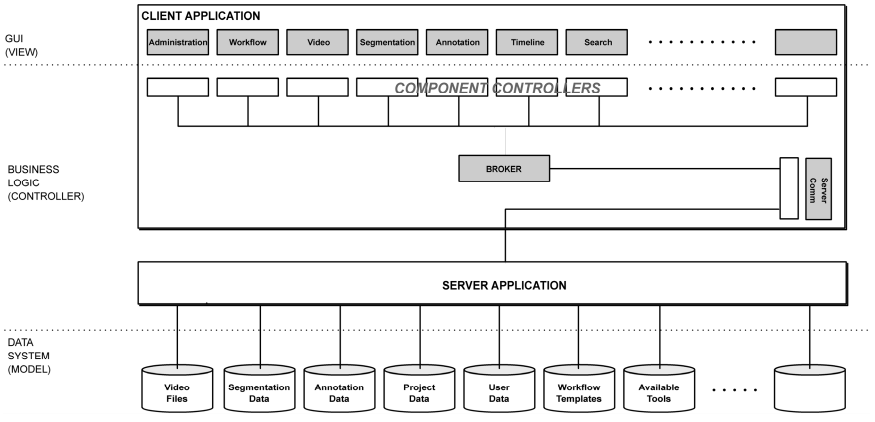

Fig. 1. Workflow-based Reference Architecture Model

for any client that is connected. Additionally, it provides several services such as authentication of annotators, and management of accounts and access rights. It is also responsible for consistent storage and management of global project configurations and video annotation information, but also workflow-related data like workflow-templates and available tools. The central application provides support for distributed authoring process, e.g. tracking of changes made by co-annotators. The client application handles user entries and interaction on the graphical user interface. It provides authoring options and assigned tools for video and annotation editing purposes.

4.2 Model View Controller and Mediator Pattern

In order to realize an appropriate management of media files and its annotated information, we rely on existing approaches with regard to video annotation or video linking. In the area of hypermedia research, several models can be identified, e.g. the *Dexter Hypertext Reference Model*. The Dexter model divides the system into three delimited layers, separating data, the given hyper structure, and its representation [13]. The *Model View Controller Model* (MVC) equally divides the application into three levels: the *model* layer represents the involved data, *views* display the information and assume user interaction, the *controller* layer processes user entries and is enabled to modify data in the model. Furthermore, data consistency is warranted through a specific notification policy [8]. In the presented workflow-based reference architecture, the model layer consists of data and information from video files, video segments and assigned annotations to project configuration information. Particularly, workflow-related data are stored workflow templates (which can be predefined by tutors) and a listing of the available tools. The view layer represents any visual component at the graphical user interface. Besides the general elements of the user interface, the single views display the available tools and methods which are previously assigned to respective tasks of the annotation workflow. The controller layer includes two different kinds of controllers: local controllers are assigned to every component of the view (as well as the server communication component) and act as interfaces between component and application. The broker component serves as global controller and implements the included *Mediator* pattern.

The *Mediator* pattern provides a central instance which defines the cooperation and interaction of multiple objects. This central unit holds an intermediary role and coordinates the overall behaviour of the system [11]. Thus, workflow control can be supported with regard to transitions between workflow phases, sequences of sub operations, passing through loops, and re-entries to other phases of the workflow. The specific processes and sequences within the annotation workflow are defined by task groups and sub operations, which can be pooled into several system components.

4.3 Task-Related Components

As mentioned above, the identified phases and tasks need to be pooled into functional units that are mutually delimited in order to typecast task areas that can be invoked by the mediator object. Furthermore, the architecture must enable administrators to integrate, replace and remove tools that can be assigned to these task-related modules. Thus, the phases of the workflow are implemented as *software components*. A software component can be seen as an enclosed unit which provides specific services. It can be embedded into a higher-level system and combined with other components. The concrete implementation of a component is concealed from its accessing instance, the communication is provided by specific interfaces [29]. Within our architecture model, the components are abstract and serve as containers for previously assigned tools and methods. Furthermore, they may be implemented several times. Thus, extensibility of the framework is supported.

Based on the identified workflow phases and tasks, we conceived and included the components *Broker*, *Administration*, *Workflow*, *Video*, *Segmentation*, *Annotation*, *Timeline*, *Search*, and *Server Communication*. In order to comprise sequences of tasks and sub operations, as well as possible loops and re-entries, we defined functions, internal procedures, and interaction with other elements of the architecture for each single component. In the following, the task-related components of the reference architecture are going to be exemplified, with a focus on their specific functions.

The *Broker Component* implements the mediator pattern and servers as global controller within the controller layer of the MVC model. It does not hold information about the concrete implementation of the system components; the communication is conducted via the components' controllers which serve as interfaces. The workflow component informs the broker instance that this task has to be performed. In order to control the whole task processing, the broker activates and highlights, or disables and hides respective components. In addition to that, it controls the interaction and communication between involved system components. Once modifying operations are performed by one component, changes must be registered in the data system one the one hand. On the other hand, other components have to be notified. Thus, the specific notification policy of the MVC model is realized. Information annotated to dynamic media like video comprises temporal conditions [9], [10], [14]. Thus, the representation of segmentation and annotation data has to be synchronized with the playing video.

The *Administration Component* is responsible for all administrative processes and configuration of the application. It provides input interfaces for configuration of general application and project properties as well as the management of user accounts, groups, roles, and access right. In addition to that, workflow- and task-related settings can be edited, e.g., creating, editing and removing of workflow templates. Workflow templates define the tasks that have to be accomplished, the operating order and

sequences, as well as the available tool(s) that are assigned to a certain task. Furthermore, distribution of tasks to different users or user groups can be stored. In general, the administration component provides interfaces for different types of information entry, and allocates data to its proper destination.

Conjointly with the broker and administration components, the *Workflow Component* forms the workflow management framework of the system. On the graphical user interface, the workflow component visualizes the tasks that have to be accomplished, and enables users to select an item in order to perform a certain task. After the selection of a task item, the broker instance is notified for workflow control purposes. Once all relevant procedures are finished, the broker notifies the workflow component about the previously selected task being accomplished. In that case, the representation of the workflow is updated.

The *Video Component* displays the assigned video file(s), as well as respective video segments and annotated data. Common interactive elements such as play, pause, stop, rewind, etc. are provided. In addition to that, further potentially services have been identified during the comparative analysis of the state-of-the-art, e.g. (synchronized) playback of multiple videos, multiple types of playback and control, or provision of keyboard shortcuts. For this purpose, the video component needs to provide appropriate interfaces. In close collaboration with the segmentation component(s), video chunking activities can be performed upon the video display area.

The *Segmentation Component* includes required video chunking approaches. Since segmentation activities are usually performed on other components like video players or timelines [17], [19], this component must provide multiple interfaces for coordination of the segmentation performance.

Concrete implementations of the *Annotation Component* enable supply, representation and editing of annotated data such as metadata, descriptions, categorization, commentary, etc. The annotation component has read access to the annotation data in the model layer in order to represent this information synchronized with the respecting video and segment(s). The selection of an annotation instance by the user must be enabled. With regard to this, the broker component has to be notified in order to initialize components like video player or timeline to update their representation of the respective data. For explorative purposes, the representation of annotated data needs to be modified. I.e., tools for grouping, sorting, filtering, etc. must be provided. This bears not only on a user's own annotated information, but also on external data like co-annotators' results or annotations within previous video annotation projects. Any modification performed upon the annotation component has to be registered in the central data system. Thereto, the broker instance has to be notified.

The conducted expert interviews and the comparative analysis of current applications revealed that segmentation and annotation activities are often performed along a *Timeline* representation [15], [17], [19]. One fundamental reason is the temporal conditions of information that is annotated to video-based media [9], [10], [14]. In addition to that, other system components may use a timeline representation of annotated information, e.g. statistical comparisons of multiple users' results.

The *Exploration Component* provides essential functionalities for browsing, searching, and comparing several kinds of information. Among this information are own results, results of co-annotators or experts, annotation data of other projects that are located

in the same data system, and also external resources. Consequently, the exploration component must provide an appropriate representation for different kinds of data.

The *Server Communication Component* is responsible for communication, transaction, and data exchange between clients and the central server application or the data layer. For this purpose, the component has to know the employed protocol(s) and support marshalling procedures.

5 Conclusion and Future Work

In this paper, we present a workflow model for collaborative video annotation processes. Based on this model, we illustrate a reference architecture that particularly supports transitions between phases and sub tasks of the (collaborative) video annotation workflow by complying the basic workflow-related requirements control, task enclosure, extensibility, and information consistency. The applied client-server model realizes co-annotator interconnectedness and information exchange by means of a centralized data system. Furthermore, services for user management, distributed authoring, and data consistency are provided. By pooling work items and tasks into enclosed software components, the regulation and control of the annotation process by a central broker instance is facilitated. An arrangement of the architecture elements along a MVC model ensures appropriate handling and management of the video files and respective additional information. Thus, it can be drawn on the presented architecture model in order to design (collaborative) video annotation software with an integration workflow management framework.

Up to date, we implemented the exemplified system architecture as well as basic software components. Elementary workflow sequences (which are under permanent further development) can be passed through. Future steps relate to the representation and user interaction referring to information displayed at the graphical user interface. This bears especially on the described workflow component and the control of the different tools that are provided on the user interface.

References

1. Baecker, R.M., Fono, D., Wolf, P.: Toward a Video Collaboratory. In: Goldman, R., Pea, R., Barron, B., Derry, S.J. (eds.) Video Research in the Learning Sciences, pp. 461–478. Lawrence Erlbaum Associates, London (2007)
2. Banerjee, S., Cohen, J., Quisel, T., Chan, A., Patodia, Y., Al-Bawab, Z., Zhang, R., Black, A., Stern, R., Rosenfeld, R., Rudnicky, A., Rybski, P.E., Veloso, M.: Creating multimodal, user-centric records of meetings with the carnegie mellon meeting recorder architecture. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, Meeting Recognition Workshop (2004)
3. Bertini, M., Del Bimbo, A., Cucchiara, R., Prati, A.: Applications ii: Semantic video adaptation based on automatic annotation of sport videos. In: 6th ACM SIGMM International Workshop on Multimedia Information Retrieval, pp. 291–298. ACM Press, New York (2004)
4. Bertino, E., Trombetta, A., Montesi, D.: Workflow Architecture for Interactive Video Management Systems. In: Distributed and Parallel Databases, pp. 33–51. Springer, Netherlands (2002)

5. Bortz, J., Döring, N.: Forschungsmethoden und Evaluation für Human- und Sozialwissenschaftler, 4th edn. Springer, Berlin (2006)
6. Brugman, H., Crasborn, O.A., Russel, A.: Collaborative annotation of sign language data with peer-to-peer technology. In: 4th International Conference on Language Resources and Evaluation, pp. 213–216. European Language Resources Association, Paris (2004)
7. Brugman, H., Russel, A.: Annotating multi-media / multi-modal resources with ELAN. In: 4th International Conference on Language Resources and Evaluation, pp. 2065–2068. European Language Resources Association, Paris (2004)
8. Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., Stal, M.: Pattern-Oriented Software Architecture. A System of Patterns, vol. 1. John Wiley & Sons, Chichester (1996)
9. Chambel, T., Zahn, C., Finke, M.: Hypervideo Design and Support for Contextualized Learning. In: IEEE International Conference on Advanced Learning Technologies, pp. 345–349. IEEE Computer Society, Los Alamitos (2004)
10. Finke, M.: Unterstützung des kooperativen Wissenserwerbs durch Hypervideo-Inhalte. Dissertation, Technische Universität Darmstadt, Germany (2005)
11. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns - Elements of Reusable Object-Oriented Software. Addison-Wesley, Reading (1995)
12. Hagedorn, J., Hailpern, J., Karahalios, K.G.: VCode and VData: illustrating a new framework for supporting the video annotation workflow. In: Working Conference on Advanced Visual Interfaces, pp. 317–321. ACM Press, New York (2008)
13. Halasz, F., Schwartz, M.: The Dexter Hypertext Reference Model. Communications of the ACM 37(2), 30–39 (1994)
14. Hofmann, C., Hollender, N.: Kooperativer Informationserwerb und -Austausch durch Hypervideo. In: Mensch & Computer 2007: Konferenz für interaktive und kooperative Medien, pp. 269–272. Oldenbourg Verlag, München (2007)
15. Hofmann, C., Hollender, N., Fellner, D.W.: A Workflow Model for Collaborative Video Annotation - Supporting the Workflow of Collaborative Video Annotation and Analysis performed in Educational Settings. In: International Conference on Computer Supported Education 2009 (to appear, 2009)
16. Hollender, N., Hofmann, C., Deneke, M.: Principles to reduce extraneous load in web-based generative learning settings. In: Workshop on Cognition and the Web 2008, pp. 7–14 (2008)
17. Kipp, M.: Spatiotemporal Coding in ANVIL. In: 6th International Conference on Language Resources and Evaluation. European Language Resources Association, Marrakech (2008)
18. Lin, C.Y., Tseng, B.L., Smith, J.R.: Video Collaborative Annotation Forum: Establishing Ground-Truth Labels on Large Multimedia Datasets. In: TRECVID 2003 Workshop (2003)
19. Link, D.: Computervermittelte Kommunikation im Spitzensport. Sportverlag Strauß, Köln (2006)
20. Mikova, M., Janik, T.: Analyse von gesundheitsfördernden Situationen im Sportunterricht: Methodologisches Vorgehen einer Videostudie. In: Mužík, V., Janík, T., Wagner, R. (eds.) Neue Herausforderungen im Gesundheitsbereich an der Schule. Was kann der Sportunterricht dazu beitragen? pp. 248–260. MU, Brno (2006)
21. National Research Council Comittee on a National Collaboratory: National Collaboratories: Applying information technology for scientific research. Nation Academy Press, Washington (1993)
22. Pea, R., Hoffert, E.: Video workflow in the learning sciences: Prospects of emerging technologies for augmenting work practices. In: Goldman, R., Pea, R., Barron, B., Derry, S.J. (eds.) Video Research in the Learning Sciences, pp. 427–460. Lawrence Erlbaum Associates, London (2007)

23. Pea, R., Lindgren, R., Rosen, J.: Computer-supported collaborative video analysis. In: 7th International Conference on Learning Sciences, pp. 516–521. International Society of the Learning Sciences (2006)
24. Ratcliff, D.: Video Methods in Qualitative Research. In: Camic, P.M., Rhodes, J.E., Yardley, L. (eds.) Handbook of Qualitative Research in Psychology: Expanding Perspectives in Methodology and Design, pp. 113–130. American Psychological Association, Washington (2003)
25. Richter, K., Finke, M., Hofmann, C., Balfanz, D.: Hypervideo. In: Pagani, M. (ed.) Encyclopedia of Multimedia Technology and Networking, 2nd edn., pp. 641–647. Idea Group Pub., USA (2007)
26. Seidel, T., Prenzel, M., Kobarg, M. (eds.): How to run a video study. Technical report of the IPN Video Study. Waxmann, Münster (2005)
27. Snoek, C.G.M., Worring, M.: Multimodal video indexing: A review of the state-of-the-art. Multimodal Tools and Applications 25(1), 5–35 (2005)
28. Stahl, E., Finke, M., Zahn, C.: Knowledge Acquisition by Hypervideo Design: An Instructional Program for Univiversity Courses. Journal of Educational Multimedia and Hypermedia 15(3), 285–302 (2006); Association for the Advancement of Computing in Education, Chesapeake
29. Szyperski, C.: Component Software - Beyond Object-Oriented Programming. Addison-Wesley, Reading (1999)
30. Volkmer, T., Smith, J.R., Natsev, A.: A web-based system for collaborative annotation of large image and video collections: an evaluation and user study. In: 13th Annual ACM international Conference on Multimedia, pp. 892–901. ACM Press, New York (2005)
31. Zhang, Q.Y., Kankanhalli, M.S., Mulhem, P.: Semantic video annotation and vague query. In: 9th International Conference on Multimedia Modeling, pp. 190–208 (2003)