This is a preliminary version of an article published by
Stephan Kopf, Stefan Wilk, Wolfgang Effelsberg
Bringing Videos to Social Media
Proc. of IEEE International Conference on Multimedia and Expo (ICME), Melbourne, Australia, pp. 681 – 686, 2012

BRINGING VIDEOS TO SOCIAL MEDIA

Stephan Kopf, Stefan Wilk, Wolfgang Effelsberg

Dept. of Computer Science IV
University of Mannheim, Mannheim, Germany
{kopf | wilk | effelsberg}@informatik.uni-mannheim.de

ABSTRACT

Although the importance of video sharing and of social media is increasing from day to day, a full integration of videos into social media is not achieved yet. We have developed a system that maps the concept of hypervideo – allowing to annotate objects in a video – to social media. We define this combination as *social video* that simultaneously allows a large number of users to contribute to the content of a video. Users can annotate video objects by adding images, text, other videos, Web links, or even communication topics. An integrated chat system allows users to communicate with friends and to link these topics to distinct objects in the video. We analyze the technical functionality and the user acceptance of our social video system in detail. Due to the integration into the social network Facebook more than 12,000 users have already accessed our system.

Index Terms— Social media, social video, hypermedia, video annotation.

1. INTRODUCTION

Nowadays, *social media* focuses on the web-based creation and exchange of user-generated data like text, Web pages, or images. Although many websites allow users to view, upload, and share videos, the editing of video content similar to the concept of the Web 2.0 is not supported yet. Especially in the case of videos, the interaction is very limited like starting, pausing, or jumping to certain points in time. For example, *YouTube* offers the possibility to add comments or upload and watch videos but advanced annotation functionalities are not supported. Entertainment aspects are the major reason for the popularity of video sharing websites. Our system pursues the same objective but also allows users to interact with the video content and provides improved functionality for communication. The combination of entertainment, communication, and video content is – from our point of perspective – a very important factor for attracting users. This assumption is supported by our evaluation and the large number of users who accessed our system.

A central *functionality of hypervideos* is the user interaction. The idea is directly derived from hypertext which uses the medium text and additional hyperlinks to refer to other Web pages. In the case of hypervideos, objects in the video refer to other content. This allows users to connect different videos or to embed other media like text or images. A major challenge is how to embed links to video objects. A manual annotation is not feasible because objects are visible in dozens or even hundreds of frames. On the other hand, automatic object tracking is challenging because objects move or change their shapes.

The remainder of the paper is structured as follows: Section 2 gives an overview on the current state of research concerning social videos. Section 3 describes the details of our social video system. The evaluation in Section 4 includes user studies as well as a technical analysis of the object tracking component. Conclusions and future work are presented in Section 5.

2. RELATED WORK

Early hypervideo systems like *HyperCafe* [1] or *Hyper-Hitchcock* [2] made it possible to access and navigate in annotated videos, but the creation of new content was usually not possible for users. These systems did not support central functionalities of social media like the interaction and collaboration of users when annotating a video.

Millions of people around the world use *social media* every day. Big players like *Facebook*, *LinkedIn*, *Twitter*, or *MySpace* reach different groups of people with different interests. *Video sharing* has evolved as a major part of social media. *HackDay* [3] was invented with the idea in mind, that a huge amount of people should annotate and share video with each other. The system allows users to share sequences of a video with each other and supports the communication between users. End-user generated content is also part of the *CWaCTool* [4] which uses *YouTube* and allows the annotation of single frames by adding audio or text. Tracking is not part of the system and thus annotations are only accessible in a limited way. Cesar *et al.* have developed a system that focuses on sharing and provides a sophisticated recommendation system [5].

Social video platforms like *YouTube* or *Yahoo! Videos* integrate more and more functionalities for *social interaction*.

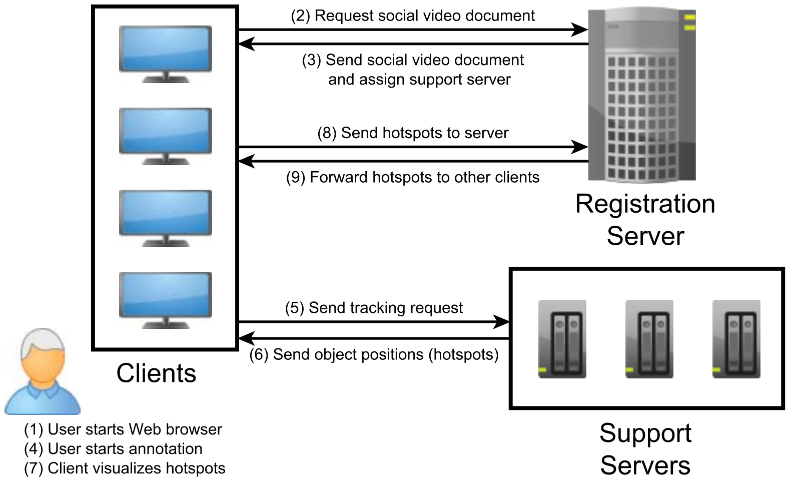
(2) Request social video document
(3) Send social video document and assign support server
(8) Send hotspots to server
(9) Forward hotspots to other clients
Registration Server
(5) Send tracking request
(6) Send object positions (hotspots)
Clients
(1) User starts Web browser
(4) User starts annotation
(7) Client visualizes hotspots
Support Servers

Fig. 1. Communication between clients, registration server, and support servers.

This allows users to annotate, to comment, and to rate videos [6]. The *Raconteur system* [7] combines the ideas of social media and real-time communication by creating a chat that makes it possible to tell a story to different users. Videos can be integrated at any point in time of the story. The idea of distance storytelling and interactive media was also realized in the *StoryVisit system* [8]. It allows adults to tell stories to children over long distances by illustrating the story with images and videos.

Current platforms as *YouTube* or *Yahoo! Videos* allow commenting and sharing of videos between friends. Still, the idea of video annotation in the context of social media is neglected. Besides static textual comments, *e.g.*, to embed subtitles to videos or to add a description to a video shot, current systems *focus on sharing* instead of supporting interactivity with video objects. In contrast to previous systems, our social video system supports the following functionalities:

1. The system has been implemented as a *distributed web-based system* which makes it accessible via standard Web browsers. This allows an easy integration into arbitrary Web pages or social networks.
2. The user interface combines annotation and navigation on one screen. This allows users to *access* and to *collaboratively add* information to videos.
3. The user interface is intuitive, easy to use, and supports a direct communication between users via chat. To support users in finding and accessing information the system includes a rich set of navigational tools.
4. The system is scalable and supports a large number of concurrent users.
5. Users are not willing to manually mark an object in all frames of a video shot. The technical infrastructure includes automatic object tracking to reduce the manual effort when a user wants to insert a link to a video object.

3. THE SOCIAL VIDEO SYSTEM

Our social video system is structured as follows. It consists of three components: a central registration server, a variable number of clients, and $0 \ldots N$ additional support servers. Fig. 1 shows the main components of the system and visualizes the communication between them.

The *central registration server* is the initial point of communication. Its main task is to refer clients to their designated support servers and merge the modifications they do to the social video documents. The main task of a *support server* is to handle object tracking requests. The central registration server starts and stops support servers and assign them to the clients. Requests for tracking an object are then directly sent from a client to the assigned support server. Our system uses *Amazon EC2 small-size instances*[1] as support servers.

The client software implements the basic concepts of our system – *navigation* and *annotation* – in one single user interface. The social video clients are responsible for bringing annotated videos to the users, allowing them to access information and to embed their own annotations. Fig. 2 shows the graphical user interface of our system. Most interactions take place in the *interactive video area*, where the social video and the hotspots are visualized. Additional information can be accessed in the *structural area* and the *dynamic information area*. An additional *options menu* allows to access the main settings of the system.

A standardized communication module based on HTTP makes the integration of the client's user interface into most Web pages and social media applications possible. The func-

[1] `http://aws.amazon.com/ec2/`

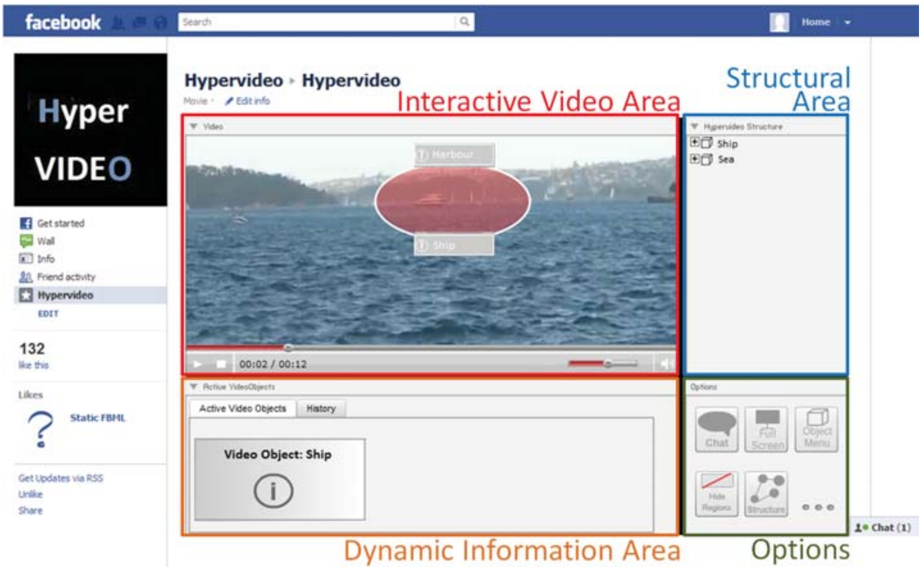

Fig. 2. Overview of the graphical user interface of the social video system. Two information nodes have been added to the video object *ship*.

tionality of the current version of *HTML5* is not sufficient to support all features of our system. Therefore, a prototype of the social video client was implemented using ActionScript 3[2], which allows the development of Flash[3] applications. This guarantees a high accessibility due to the integration of the client into standard Web browsers.

3.1. Navigation Interface

To make information accessible, a *hotspot* is visualized as a rectangular overlay of the annotated video object. Users can directly interact with the object by clicking on its hotspot. The system then pauses the video and displays the *information nodes*. The system allows to access different information nodes, like *images*, *text*, links to *Web pages*, *videos*, and *communication topics*. When activating a Web node, the referred Web page is opened in a new tab or instance of the Web browser. All other information nodes are immediately visualized in the interactive video area.

Another possibility to access information is to use the *video object tree* (see Fig. 3, left) that shows all video objects of the current video. It differs from hotspots because the tree makes it possible to access object information at any time, independent of the motion, size, and visibility of the annotated object. The *structural view* was designed to avoid that users become lost in the hypervideo space (see Fig. 3, right). It shows all linked videos as a graph. A user can retrieve his/her current position in the hypermedia space and see possible navigation paths. When the user moves the mouse to a node, a preview of the video is started. By clicking on a node, the system jumps directly to the video and starts playing.

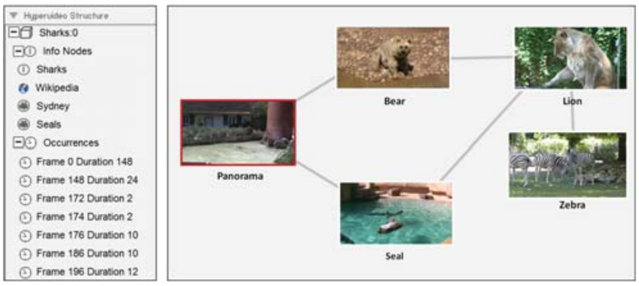

Fig. 3. Determining the current position in hypervideo space: Object tree (left) and structural view (right).

3.2. Annotation Interface

The *annotation* – also called *authoring* – is a central concept of our social video system that allows every user to bring his/her own ideas and knowledge into the video. An important concept is to understand annotation not as a centralized but as a distributed and, at the same time, social process. Such a social process allows multiple users to work on the same video document at the same time and share the annotations they have created. This turns our video system into a platform for exchanging knowledge in a social way. To motivate users not only to consume information but also to participate in creating new content, the annotation process should be *intuitive* and *very easy to use*.

Creating a new video object is easily done with the mouse

[2] http://www.adobe.com/devnet/actionscript.html
[3] http://www.adobe.com/products/flash.html

by clicking on the interactive video area and by drawing a rectangular region. The object in the current frame is marked, and the first step of the annotation process is finished. Already at this point, the initial marking of the object automatically invokes the server-based object tracking. The tracking algorithm does not stop at shot boundaries but processes all frames of the entire video. In a second step, users are able to *connect additional information* to a video object. Multiple links may be added to one object. The system offers four types of links which are called *information nodes*: Web nodes, image-text nodes, video nodes, and communication nodes.

Web nodes create links to other Web pages, and *image-text nodes* combine pieces of static text and images. The most important nodes are *video nodes*: Every linked video can automatically be used as a social video, and users can immediately annotate it after the upload and the automatic pre-processing by the server. Communication between users is supported by a special *communication node* that users can link to video objects. Communication nodes generate new topics in the integrated chat system, allowing users to exchange questions, ideas, or comments. All communications are stored in the system so that any user can access and continue the communication at any time. The chat system also integrates other social media and supports the communication with Facebook users via *Facebook Chat*[4] using the *Extensible Messaging and Presence Protocol*[5] (XMPP). This integration makes it possible to directly invite friends to participate in our social video system.

3.3. Object Tracking System

Interactive regions in our social video system are visualized using *Hotspots*. By simply marking a region within a video frame (clicking and dragging the mouse) a video object is defined. To limit the manual overhead for users an automatic tracking system uses this region as a template to be tracked. We known from experience that it is not possible to get robust results in different videos by using only one object tracking algorithm. Our idea is to analyze the *properties* of the *template* and the underlying *video* to automatically select the algorithm that works best. Our system decides dynamically which of the following three tracking algorithms is used: MeanShift [9], template-based matching, and Speeded Up Robust Features (SURF) [10] combined with the Kanade-Lucas-Tracker (KLT) [11].

The idea of *MeanShift* is to identify regions with characteristic colors. *Template-based matching* is a brute-force approach that compares color or intensity values between a template and a series of consecutive frames. In earlier work, we used Harris feature points to estimate the camera motion and to segment moving objects [12]. We substituted Harris for SURF [10] that is more robust to scaling and rotation. The idea of KLT is to calculate the pixel displacements of consecutive frames on the basis of motion vector fields.

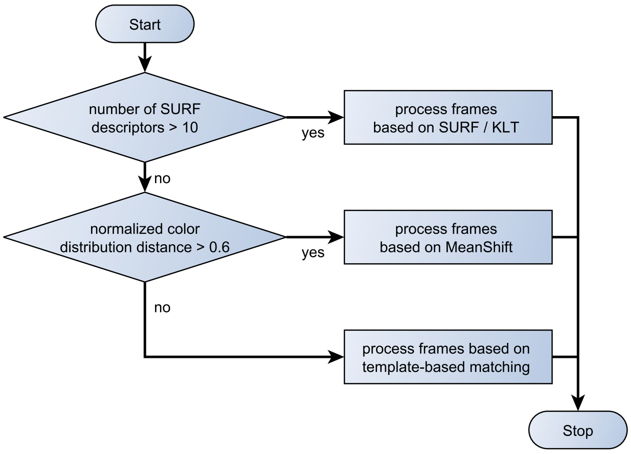
Start
number of SURF descriptors > 10
yes
process frames based on SURF / KLT
no
normalized color distribution distance > 0.6
yes
process frames based on MeanShift
no
process frames based on template-based matching
Stop

Fig. 4. Selection of a tracking algorithm.

Fig. 4 gives an overview of the selection process. For example, a template can only be tracked when it contains a sufficient number of SURF descriptors. Our algorithm uses SURF matching if at least ten valid descriptors are located in a template. *MeanShift* is selected if the color distribution of the template is different from the color distribution of the shot. The average precision of MeanShift is significantly lower compared to SURF feature matching, but it is still robust against scaling and rotation. In all other cases *template-based matching* is selected. It can always be used but it is very sensitive when objects are scaled or rotated. More details about the tracking algorithm have been published in [13].

A technique our social video system uses to reduce the load caused by multiple users is to *distribute the object tracking task to different machines*. New support servers using *Amazon EC2 small-size instances*[6] are automatically started as necessary. During the evaluation, an assignment rate of 20 users to one *Amazon EC2 instance* proved to be sufficient.

4. EVALUATION

The evaluation of the social video system consists of a technical analysis of the automatic object tracking as well as two independent experiments with users. Facebook was chosen as prototyping environment for the evaluation. Fig. 2 shows the integration of the social video system into Facebook. This approach attracted more than 12,000 users[7] within four months. Users answered questions about the interaction with the system, *e.g.*, how users access information, embed their own ideas, or collaborate with each other.

[4] `http://developers.facebook.com/blog/post/361/`

[5] `http://xmpp.org/`

[6] `http://aws.amazon.com/ec2/`

[7] The number of individual visitors of the social video system was counted by Google Analytics, `http://www.google.com/analytics/`

Table 1. Test users and their attributes: Knowledge ranges from 1 (very low) to 5 (very high); the variance is specified in brackets.

Type of Evaluation	Navigation	Annotation
Content of social videos	zoo	cocktails
Number of probands	75	225
Avg. Age	30.33	27.01
Gender		
- female	29	68
- male	46	157
Knowledge of		
- Movie Editing	1.65 (0.85)	2.24 (1.09)
- Hypermedia	1.85 (0.88)	2.54 (1.45)
- E-Learning	2.61 (1.78)	2.32 (1.57)
- IT	3.18 (2.01)	3.28 (1.31)

We have developed a *survey module* that is fully integrated into the social video system as part of the interactive video area. Groups of questions were gathered to identify strengths and weaknesses of every system feature. When a user activates a feature, the system triggers the survey module to show the corresponding group of questions to the user and to allow a rating. Different questions based on a 5-step Likert-scale [14, p. 247] make it possible to determine the effect, the advantages, and the disadvantages of different aspects of the system. A data set was only considered for evaluation if a user answered all 48 questions.

4.1. Accessing Information in the Social Video System

Accessing embedded information and linked videos is one of the most important features of our social video system. We use the event-based survey module to collect the data how users access the system. During the first evaluation experiment, 75 probands evaluated different features of the social video system. The questions focus on the interaction with hotspots, how users access the information in videos, as well as questions concerning the practical use of the system. The pre-knowledge of the test users concerning hypermedia, E-Learning, and information technology varies significantly (see Table 1). Videos and additional information about the topic *'zoo'* have been selected for the user evaluation. Fig. 3 (right) shows some video clips that are used in this evaluation. The video clips include eleven video objects in total.

Hotspots are the main form of interaction for users. Since the position and size of a hotspot may change during the playback of a video, the first question is to determine how many hotspots should be visualized simultaneously. The majority of probands (88%) accept three or more hotspots at the same time. Users describe that multiple hotspots are accepted as long as they do not occlude each other. 52% of the users identified this as a potential problem during navigation. When interacting with a hotspot, most users agree that pausing the video is a good or very good idea (96%). In general, hotspots are easy to use with an average value of 4.06 (1: very hard to use; 5: very easy to use).

The benefits of the **additional navigation tools** – like the *video object tree* and the *structural view* – have also been evaluated in this experiment. The structural view is mainly used to navigate between videos on a high level whereas the video object tree allows to access details in a video. With an average value of 4.1, the user interface was evaluated fairly good. Especially the video object tree and the structural view were graded as very useful and easy to use. More than two thirds of the users gave feedback that both tools helped significantly in retrieving the information that they wanted to get (ratings of 4 or 5). Especially the combination of video object tree and structural view makes a fast navigation possible (avg.: 4.24; variance: 0.32). The acceptance of the integrated, event-based survey module was evaluated, too. Only 2 of 75 users rated it as disturbing during the usage of the system.

4.2. Adding Annotations in the Interactive Video Area

In a second evaluation, 225 users (see Table 1) tested the *annotation capabilities* and especially the *combination of annotation and navigation* in our social video system. A tutorial video was provided for users to become familiar with the annotation process. The task of the users was to annotate the videos and to give feedback on the ease of use of the annotation functionalities. When a new type of information node was added, the integrated survey module asked for feedback. Several short video sequences have been added to the system which explain different steps *how to mix cocktails*. Some complete cocktail recipes have also been made available as social videos. Users had the possibility to annotate these clips and, of course, to connect them to create their own cocktail recipes. The embedding of our system into social media made it possible for users to add their own video clips and share them with other users. By uploading videos and combining them with existing clips, users created 23 new social video sequences that can be classified as real cocktail recipes.

The annotation process is very intuitive for users and it motivated all users to annotate videos and combine them into new recipes. Table 2 shows that only a small percentage of users rate the **creation of a new information node** to be complicated or very complicated. An overall number of 1,148 annotations were added to the system.

An important aspect of social media is the **communication with other users**. We have integrated a chat application as well as the API of Facebook to facilitate a discussion about the content of a video. 96 communication nodes were initiated in the system that included 56 questions. All questions were answered or discussed in the evaluation period. This makes it clear that the users are highly motivated to help each other. On average, each communication node and thus chat topic is linked to 4.1 users (variance: 2.1).

Table 2. Number of users that rate the difficulty to create new information nodes between 1 (very complicated) and 5 (very easy).

Value **Node**	**1**	**2**	**3**	**4**	**5**	**Average**
Text-image	9	6	65	77	68	3.84
Video	11	28	141	33	12	3.03
Communication	0	15	80	66	64	3.80
Web	6	18	72	61	68	3.74

4.3. Automatic Object Tracking

The main task of our adaptive object tracking is to correctly position the hotspots. The following evaluation measures whether this adaptive approach based on different algorithms is able to handle complex situations in different video sequences.

Precision and latency of the object tracking algorithm are the most relevant requirements for an efficient usage of our system. **Precision** measures the quality of detecting the correct position of a video object. Deviations may lead to misinterpretation or confusion as the user is no longer able to identify what the hotspot annotates. **Latency** determines the period of time the algorithms need to calculate the position of a hotspot in the entire video sequence. High latencies may lead to the same negative effect as low precision. When the user resumes a video after completing the annotation, he/she expects the new hotspot to move according to the object motion.

We selected a set of 16 video sequences which include situations where the tracking has to handle changes in illumination, deformation of objects, and occlusion. Videos with many scene breaks and different resolutions down to 320 x 240 pixels complete the test set. Low resolution videos are especially challenging as details are lost and thus the precision of feature-based approaches drops. As ground-truth data and a basis for the evaluation of the adaptive tracking, 2,526 reference object positions in randomly selected frames were marked manually in the video sequences.

We compare the adaptive tracking algorithm to the performance of SURF feature matching, the MeanShift algorithm, and template-based matching. Tracked positions and object sizes are labeled as correct if the coordinates of the corners are located in a radius of 15 pixels of the reference corners. The **precision** of our *adaptive tracking* (88.16%) shows a superior performance with an improvement of 7.56% compared to *SURF/KLT feature matching* (80.60%). The more simple algorithms like *MeanShift* (30.09%) and *template-based matching* (27.36%) perform significantly worse. The **false hit rate** gives an overview on the percentage of frames in which a tracking algorithm calculated the wrong position in comparison to the overall number of frames. The adaptive tracking (7.0%) generates the lowest rate of erroneously detected positions, followed by pure SURF/KLT (7.69%), pure template-based matching (20.37%), and pure MeanShift (26.06%).

The number of concurrent users in combination with the low computational power of the small EC2 instances lead to the decision to analyze only every 6th frame of a video (assuming a frame rate of 25 – 30 fps). User experience showed, that this reduces the **latency** and improves the responsiveness of the system without limiting the quality of the results. Object tracking in real-time is now possible and the algorithms require a computation time per video second between 0.51 and 0.87 seconds depending on the test system[8] used.

The survey module asked users to evaluate the **quality and effectiveness** of the automatic tracking. The users started 383 tracking requests by defining new video objects with the mouse. 16 users (4.18% of the tracking requests) recognized some incorrect object positions. These errors typically occur in case of occlusion, object deformation, or when using small templates. Despite the incorrect object positions in some frames, the overall system was rated very good which could also be recognized in the large number of Facebook recommendations. Although a large number of users tested the system, speed limitations caused by the tracking algorithm have not been reported by any user.

5. CONCLUSIONS AND OUTLOOK

We have presented a social video system that allows users to navigate and to collaboratively annotate videos. Accessing the system is easy for users, as it can be embedded into any website and thus can be run via standard Web browsers. The extended client-server model is scalable and supports large numbers of users to collaboratively work on social videos. An intuitive interface guarantees a high user's acceptance. Especially the automatic object tracking is seen as a high benefit for users, which makes the social video system applicable in practice.

In future work, we would like to use the API of current social networks for accessing, uploading, and integrating images, videos, and social connections into our system. This would dramatically increase the amount and quality of content that can be easily integrated into our system.

6. REFERENCES

[1] Nitin Sawhney, David Balcom, and Ian Smith, "HyperCafe: Narratic and Aesthetic Properties of Hypervideo," in *Proc. 7th ACM conference on Hypertext*, 1996, pp. 1–10.

[2] Andreas Girgensohn, Frank Shipman, and L. Wilcox, "Hyper-Hitchcock: Towards the Easy Authoring of Interactive Video," in *Human-Computer Interaction*, 2003, pp. 33–40.

[8] Test systems used to measure latency: Intel Core I5 (4 x 2.27 GHz, 4 GB), AMD Opteron (2.33 GHz, 1 GB), and Intel Xeon (2 x 2.66 GHz, 8 GB).

[3] David A. Shamma, Ryan Shaw, Peter L. Shafton, and Yiming Liu, “Watch what I watch: using community activity to understand content,” in *Proc. Workshop on multimedia information retrieval*, New York, NY, USA, 2007, MIR ’07, pp. 275–284.

[4] Roberto Fagá, Jr., Vivian Genaro Motti, Renan Gonçalves Cattelan, Cesar Augusto Camillo Teixeira, and Maria da Graça Campos Pimentel, “A social approach to authoring media annotations,” in *Proc. 10th ACM symposium on Document engineering*, 2010, DocEng ’10, pp. 17–26.

[5] P. Cesar, D. Bulterman, J. Jansen, D. Geerts, H. Knoche, and W. Seager, “Fragment, tag, enrich, and send: Enhancing social sharing of video,” *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 5, pp. 19:1–19:27, August 2009.

[6] Liao Zhuhua, Yang Jing, Fu Chuan, and Zhang Guoqing, “A Tracking Model for Enhancing Social Video Integration and Sharing,” *2010 10th IEEE International Conference on Computer and Information Technology*, pp. 1571–1576, June 2010.

[7] Pei-Yu Chi and Henry Lieberman, “Raconteur: Integrating authored and real-time social media,” in *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI ’11*, New York, USA, May 2011, pp. 3165–3168.

[8] H. Raffle, J. Go, M. Spasojevic, G. Revelle, K. Mori, R. Ballagas, K. Buza, H. Horii, J. Kaye, K. Cook, and N. Freed, “Hello, is Grandma there? Let’s Read!,” in *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI ’11*, New York, USA, May 2011, pp. 1195–1204.

[9] D. Comaniciu, V. Ramesh, and P. Meer, “Kernel-based object tracking,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, May 2003.

[10] Herbert Bay and Tinne Tuytelaars, “SURF: Speeded up robust features,” in *Proc. European Conference on Computer Vision*, Graz, Austria, 2006, vol. 3951, pp. 404–417.

[11] B.D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in *International joint conference on artificial intelligence*, Acapulco, Mexico, 1981, vol. 3, pp. 674–679.

[12] Dirk Farin, Thomas Haenselmann, Stephan Kopf, Gerald Kühne, and Wolfgang Effelsberg, “Segmentation and classification of moving video objects,” in *Handbook of Video Databases: Design and Applications*, Borko Furht and Oge Marques, Eds., vol. 8 of *Internet and Communications Series*, pp. 561–591. CRC Press, Boca Raton, FL, USA, 2003.

[13] Stefan Wilk, Stephan Kopf, and Wolfgang Effelsberg, “Robust tracking for interactive social video,” in *Applications of Computer Vision (WACV), 2012 IEEE Workshop on*, jan. 2012, pp. 105 – 110.

[14] Carl D. McDaniel and Roger H. Gates, *Marketing research essentials*, Taylor & Francis, Cincinnati, 1. edition, 1998.