

OPTIMIZING THE VIDEO EXPERIENCE IN MOOCS

Jan Renz, Matthias Bauer, Martin Malchow, Thomas Staubitz, Christoph Meinel

Hasso Plattner Institute, University of Potsdam (GERMANY)

Abstract

Massive Open Online Courses have gained more and more popularity in the recent years. Video Content contributes a vital aspect of the learning experience in MOOCs. The paper at hand proposes ways to optimize the video experience in MOOCs. Single stream videos will be considered as well as the openHPI's dual stream video player. openHPI is the MOOC platform of the Hasso Plattner Institute, providing MOOCs to thousands of users since 2012. One of the unique features of our video player is the possibility to play two synchronized video streams. Based on collected usage data of our html5 based video player we evaluate the learners acceptance of features, such as adaptive playback speed, dual video scaling, full-screen mode, slide navigation and subtitles. Furthermore, we will discuss the impact on the users learning outcome.

Keywords: MOOCs, Videos, Subtitles, User Contributes Content.

1 INTRODUCTION

Videos are one out of the three most important elements in MOOCs (videos, self-tests, community). The paper at hand gives analyzes the experiences we made with our video player on openHPI and openSAP. Both MOOC platforms are operated by a research team at the Hasso Plattner Institute. While openHPI offers courses derived from the on-site course curriculum, openSAP is an enterprise MOOC platform, offering courses for SAP employees, customers, partners, and the interested public. Both offers are free to use for everybody. openHPI launched its first course in September 2012, openSAP followed in May 2013. The total number of enrollments has exceeded 900.000 recently.

1.1 Video setups

The video production for our MOOCs usually happens in the weeks before the course start and may still be done after the course start to enhance the course content, for example to give a feedback video or publish an additional tutorial. So it is crucial to have an easy working production system and workflow.

Different MOOC providers use different setups for producing their video contents. At openHPI for example, in the very beginning, the courses were recorded in a lecture hall with the help of the tele-TASK recording system¹. Later, a small studio was set up, offering a counter, an interactive whiteboard, a green screen and an adaptable light setup. Which of the offered tools are used is up to the lecturer. They have to feel comfortable and can decide themselves if they would like to use a laptop or a tablet computer with a pen or the interactive whiteboard. Of course the setup also depends on the content to be presented and the targeted audience. After the recording, an assistant will upload the learning units that have been recorded with the tele-TASK recording system to Vimeo, which is used as a global CDN. The teaching team will then create the learning units on the MOOC platform and include the Videos from Vimeo.

At openSAP an additional post-production step is added. The units are often enhanced with other learning material, such as animations. This extends the material's quality but comes for the price of significantly higher production costs.

openHPI on the other hand enhances the videos with an automatically extracted slide navigation. [4]

All these steps are invisible for the learners. Both platforms allow different layouts for video learning units. The standard layout in openHPI is a dual-stream player, which will be discussed in chapter 2. openSAP, on the other hand, makes use of a single stream player.

¹ tele-TASK Lecture Recoding Technology, <http://www.tele-task.de/teletask/about/>, last accessed on May 20, 2015

1.2 Video duration

Two of the major success factors of MOOCs are the consumability of the content, including the video content, and the immediate assessments. For both of these features short videos should be preferred. Based on the videos used in all courses run so far a detailed analysis has been conducted. Apart from some outliers (such as outtake videos used at the end of the course and some rather long videos) the typical video length is in between 5 minutes and 23 minutes (see Fig.1, left). It is notable that among the courses with a very low average video runtime are those dealing with UX (made by UX aware teams) and those with intense hands-on coding assessments, which all have very good success ratios. While there is no clear indication that shorter videos are helping to increase the course success rate, successful courses and those focussing on good learning experience on openHPI and openSAP often feature short videos (see Fig. 2).

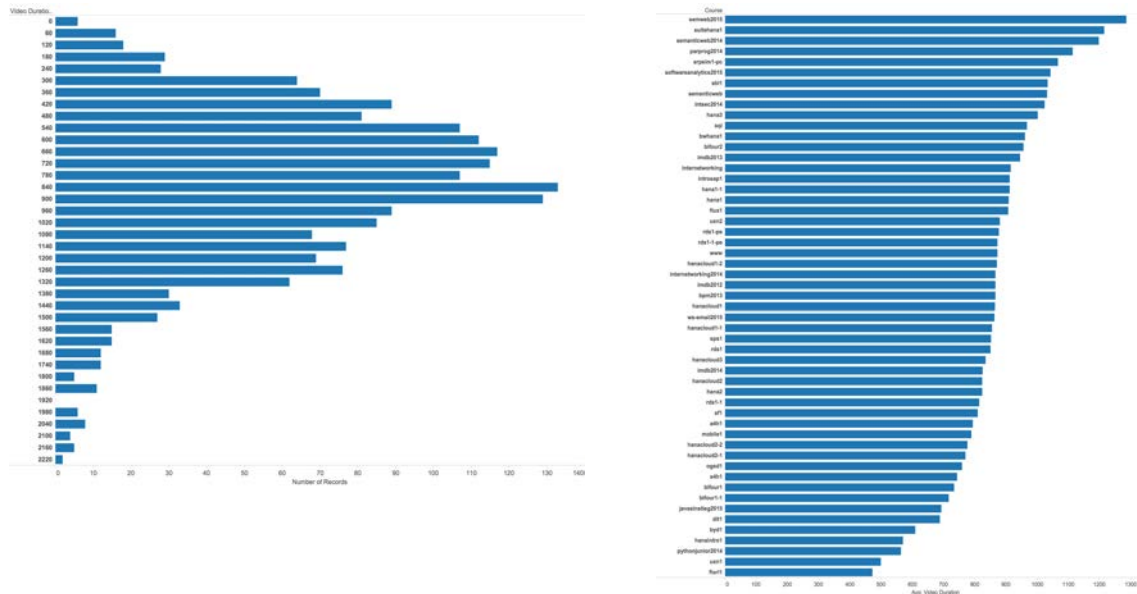


Fig. 1: Video runtime (in seconds) of all videos in openHPI and openSAP (left) and avg. video runtime per course

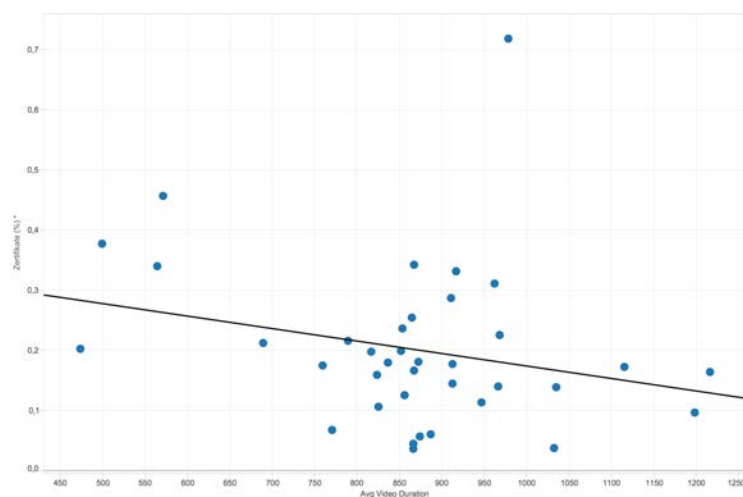


Fig. 2: success rates of a course and the avg. video length

2 PLAYER

To deliver the videos a custom HTML5 based video player with the capability to play both single and dual video streams was implemented. This player is showing the lecturer in the left stream and the slides on the right. These two streams are played in two separate HTML5 video elements

synchronously. To make that work, a JavaScript is synchronizing the time of both video screens if the time difference is more than a second. Every synchronization may lead to a very short flicker in some browsers, so the synchronization events have to be kept to a minimum. For our purposes a difference up to one second is precise enough. Only the lecturer video has an audio track. The slide video does not have an audio track and, by its nature, has only very little motion. Due to these facts no real-time synchronization is necessary for this use case and a delta in between the playback position up to one second seems feasible.



Fig. 3: The HTML5 based dual stream player with active slide nav used in openHPI

The player offers the following features for the students. Both video streams are resizable. If the user is more interested in the lecturer stream she can zoom it and decrease the size of the slide stream, thereby, automatically. Hence, the available space of the player can be used efficiently. This also works vice versa when the student wants to maximize the slide stream. It is also possible to change the playback speed and thus allowing to adjust the speakers pace to the student's. This also increases the accessibility of the videos for non-native speakers or people with certain disabilities. A feature that distinguishes our video player from other players is the slide preview. This slides are generated automatically from slide stream. The slide transitions are detected automatically and an image is saved shortly before the slide transition is performed [4]. An additional timeline navigation element is located above the video timeline. This highlights the slides shown over the time. While hovering the mouse over a specific position in this timeline the image of the slide content at that position will be shown (see Fig. 3). This enables students to search and jump fast to the slide they are interested in. Finally, the player also supports full-screen mode for both, single and dual stream, video player. All previously described features of the video player remain accessible while its running in full-screen mode.

The player also features keyboard shortcuts. These shortcuts allow to switch to full screen mode (f), toggling play and pause (p), and navigating on the timeline with the arrow keys. This function is rarely used because it is not available through an UI Element. To make sure the users are aware of non-obvious features they need to be informed. On the other hand, all secondary textual information that is displayed to the user may distract him/her and disturb his/her learning experience.

While this could be done by an onboarding tour as described by Renz et al in [1], a sleeker approach to solve this conflict is to display a hint. This hint can be hidden by a simple user interaction ("do not show anymore"). The hidden state of the hint will be saved in the user settings. Therefore, the user will only see this hint once. For hints that need more space and therefore are more distracting or for hints on features that are already used by many users the distraction impact can be minimized by a prior tracking of the relevant user interaction (in this case using any keyboard command in the player) followed by automatic disabling of this hint by default for all those users where a previous usage have been tracked.

2.1 Enabling self tests inside the player

Self tests are a useful tool for participants to review their understanding of the videos' or other materials contents [9]. On openHPI/openSAP self tests are offered after watching the video. This

allows for more detailed revisions of the participants' knowledge. Additionally, for longer videos, in-video quizzes can offer extra benefits for the participants [3]. They are closer to the topic that is being tested. They also can prevent participants from sleeping through the video. Using the in-video quizzes as milestones within a video stops the video at a certain position and thus wakes up the participants. Particularly, professional participants are prone to falling asleep after a long days work. If the student realizes that he missed the important part, she can jump back some minutes and can try to understand the missing information by reviewing the specific section only. This approach is adding additional quiz information to the HTML5 player and extending its functionality. When quiz information is found in the HTML source code for a MOOC video, the player is analyzing the timestamp, question type, and the question. According to this information, the question will be shown in the suitable layout and the right point of time within the video. When the timestamp of a question is reached, the question will be shown as overlay over the video and the video is paused automatically. If the user jumps to a later point in time and several questions occur during this duration all questions will be shown to the students one by one. Now the student has time to answer the question. After answering, the right answer will be shown and the student can analyze if she was able to answer everything correctly and can go on watching or review the critical part. Furthermore, the self tests in the HTML5 player can be disabled and enabled. This gives students the freedom to decide which learning experience is suitable for their needs. As a standard this functionality should be enabled to popularize this feature amongst the participants.

While this is an interesting feature in terms of decreasing the amount and the time in between content delivery and self-test, this will add a new layer of complexity to both the technical side (player) and the Usability of the course.

2.2 Offline Video Usage

In a UX survey conducted in spring 2015 among users of the openHPI portal 71% of the answered that the possibility to download the course material is among their most-liked features. 85% answered that they already used one of the offline features like downloading videos or slides. 57% answered that they would like to watch the videos without any internet connection. This clearly indicates the users demand for offline video support.

But if users just download the videos to watch them offline (or to only hear them, for example in the car stereo) breaks the most basic didactical concept of MOOCs of watching videos and immediately check their gained knowledge in the following self-test.

In a prototype developed by the openHPI it was shown that modern browsers are already able to store all data needed to display a course within the browser, utilizing LocalStorage and IndexedDB. In a prototype a travel mode was introduced, that allows watching videos and taking self tests offline, after a previous explicit "download". Currently further research is done to enable this feature y default without any explicit user interaction.

2.3 Mobile Video Usage

After the iPad App released in 2014 that Renz et al. described in [5] the openHPI Team also released a native android app. This app enables mobile access to all course material and offline access to slides and videos, while other mobile MOOC apps like the edX mobile app only features videos.

While the openHPI app provide one click download of videos (or even all videos within a section), there are some limits. Support of dual stream video playback is limited on all major mobile platforms. Furthermore, there is a fixed playback speed of 1x on androids internal video player. Further research is needed to enable native features like Chromecast support and identify good ways to handle dual stream display with multiple displays.

3 EMPOWERING VIDEO USAGE THROUGH TEXT-BASED CONTENT

3.1 Search

OCR transcriptions can be used for an efficient search in MOOCs videos containing lecture slide information for content analyses like it is realized in tele-TASK [4]. Especially in cMOOCs it would be reasonable to search all video material with an efficient full text search to find the videos and of interest. Furthermore, also xMOOCs achieved courses can be reused as knowledge base for students

when they are efficiently searchable. The idea of this approach is to analyse all videos with an OCR detection algorithm [4]. After this analysis, all data OCR data will be saved and indexed in the in-memory database HANA [2]. The usage of HANA is necessary to perform a full text search in a reasonable time. For an amount of 5500 videos, the time for a search request is approximately 300ms. Due to, OCR data is not completely accurate like visible in Fig. 4, which is showing the analysed data of the lecture slide (left) and the lecture slide (right) small mistakes have to be eliminated. As a solution, a fuzzy search is performed during a full text search. Because of the fuzzy search in combination with the OCR data, the small mistakes can be rescinded. This approach also reveal matching video result even when the student search misspelled words. Furthermore, this approach offers several search options for MOOC videos like "video only", "sections of videos", and "dive into the topic" [3]. These options offering different behaviour of the search to fulfil students' needs for searching appropriate videos. "Video only" is searching with the HANA and the containing OCR data of the videos and ranking the results of the found videos. The second option "sections of videos" is finding the sections where the search term occurs additionally to the video search. Finally, "dive into the topic" is finding also videos with connected topics of students' interest. For example, when searching for TCP also UDP will be highlighted by the search. This option gives students the possibility to obtain an overview and connections of a topic in other context. To realize this search option semantic web technologies will be used.

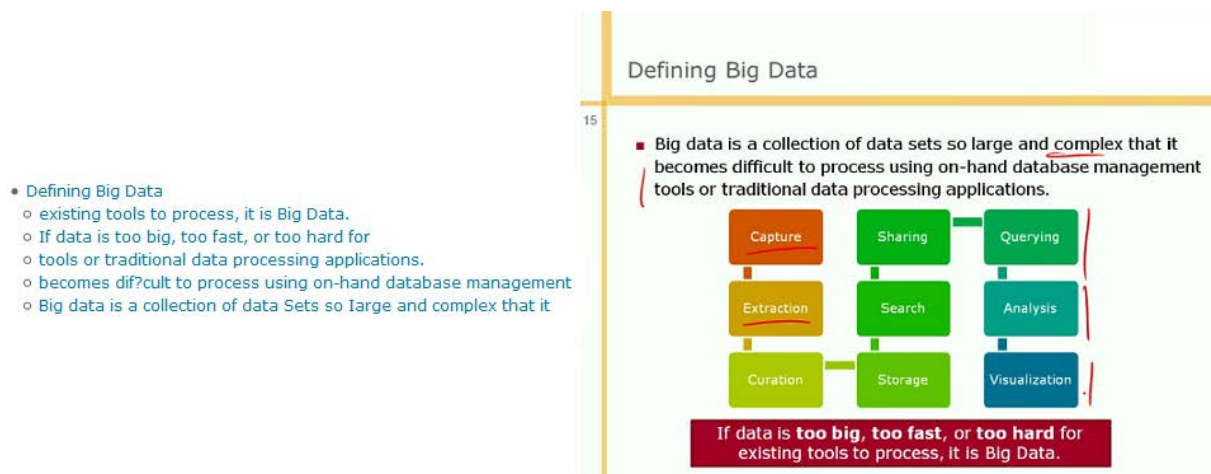


Fig. 4: OCR result for a slide of the dual video stream

3.2 Creation of subtitles / annotations

While subtitles can be created automatically using a similar technology as presented above, but instead using Audio recognition, even if this data is optimized by using the slide data its currently not really usable for subtitles without a dedicated review.

Courses on openHPI and its sister platforms are usually offered in one language (currently English, German or Chinese). With the rollout of openHPI.cn, the Chinese version of openHPI, there was a demand for annotations of the video. Also SAP introduced a localized version of their enterprise MOOC platform in China, different variants of localization have been tested. All courses offered on the Chinese Versions have been localized versions of courses that already run at least once in English or German. In all courses the reading materials are provided in Chinese. Also the Forums main language is Chinese. Videos have been rerecorded with a native (Chinese) speaker. In other courses the original videos (English) have been subtitled in Chinese. Based on the learners feedback it became clear that most learners in Chinese, at least in the context of the IT related courses, provided on these platforms preferred English speakers with subtitles.

So a concept is needed to allow easy creation and maintenance of transcriptions and subtitles in various languages.

There are several (web based) tools available that can fulfill this task that have been evaluated together with a team of students. For example YouTube offers a video transcription tool for all videos hosted on YouTube. Amara (amara.org) offers a good online subtitling tool. Udacity is using this provider and claims to have 1.600 volunteer members [6]. Amara is Open Source published under the AGPL.

This leads to a broad range of subtitles available within many courses and proves that the community is willing to engage and provide content.

Another possibility is to integrate the subtitling tool within the MOOC portal. This has several advantages:

- No additional login used
- Deep integration into the system

Esp. the deep integration enables some promising use cases:

If the system knows the user is from Spain, and there is a video that has none or an incomplete Spanish subtitle ask the user to help or grant the users who submitted translations with gamification points or badges.

A prototype of an integrated subtitles tool has been created by a group of students as part of a seminar. This prototype, called UberCaptions consists of two parts: a JavaScript based editor and a RubyOnRails Application that will server as the backend for the editor.

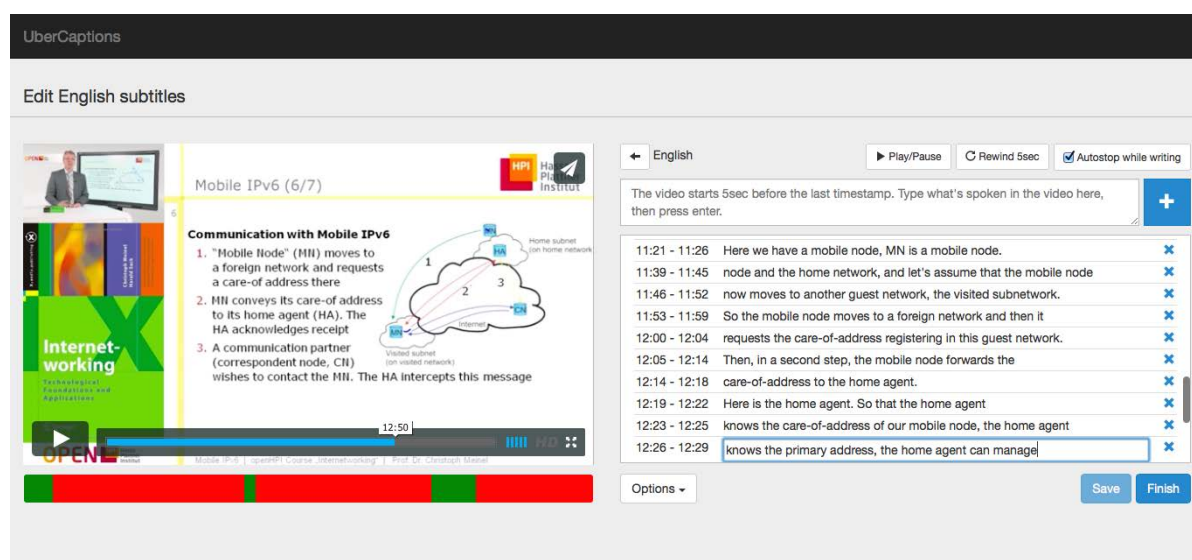


Fig. 5: Screenshot of UberCaptions.

The UI features a video player, a progress bar that highlights edited and non edited sections of the video, an input text field and a scrollable list of all created subtitles for that video in the current language.

Yousef et al [8] are evaluating different annotation tools and introduce their annotation tool Video-Mapper, that provides collaborative video annotation for their L2P-LMS. While there is the clear need to provide an annotation tool, it may depend on factors like amount of video material (as providers like Amara charge per minute in their enterprise edition) if a custom or on ready-made tool should be chosen.

4 VIDEO ANALYTICS

To fully understand the impact of video material provided within MOOCs its important to have a good understanding of the users interaction and consumption. If a MOOC provider uses the embedded player offered by video platforms like you tube or Vimeo these platforms will automatically create usage stats that can be accessed within the backends of these portals. This analytics come with several drawbacks:

- They can be only used if the standard player provided by those platforms is used (so thereby no custom features can be added)

- Usage data will be sent to those platforms, which probably will collide with data privacy regulations, especially if the MOOC provider is located in Europe as the major video providers are located in the united states
- Usage data cannot not mapped to the users data
- Usage data is not analyzing usage within the video

So in terms of learning analytics it is recommended to collect a more detailed set of data. This data can then be used to identify issues with the provided videos.

In openHPI and the other Xikolo based MOOC portals all video interaction data is tracked by a dedicated learning analytics service. Based on the concept of Activity Streams, all video interactions are using the Experience API data model, so all interactions are saved in the form: <Actor> <Verb> <Object>, with <Result>, in <Context> The Actor, usually a student in a learning platform, performs a certain action, i.e. Verb, on a specific Object (the video). With this expressive structure, it is possible to track any kind of video activity, for example “<Student John Doe> <watches> <Internet Security Video> in <1.5 playback speed>”.

The gathering, collection and processing of usage data is different based on the type of interaction. All user interactions that also trigger a server processed action can be triggered on the server side. All client side actions are handled on the client and then send to the server for further processing. Video events that are tracked include video play, video pause, switching to full-screen, resizing the dual player ratio, and changing the playback speed.

Learning analytics of video usage can be done with a short term motivation or with a mid to long term motivation. In the first case learning analytics should lead to a benefit for either the learner or the teaching team.

4.1 User acceptance and possible improvements

To determine if the features available in the current player are used by learners the learning analytics described in chapter 4 has been used. The numbers shown here are based on the tracked user interaction of recorded in may 2015.

There for unique video plays (unique by user and video item) are considered. For each of this unique plays the learning analytics data is used to evaluate the acceptance of the advanced player features introduced in chapter 2.

	Full-screen	Resize	Speed Change
Total	24.61%	-	10.75%
Single Stream	22.20%	-	10,47%
Dual Stream	39.77%	20.75%	12.54%

The data collected above give some interesting insights:

Concerning the resizing 1 out of 5 video plays contain at least one resize. This indicates that the current default player is not satisfying for many users. To improve this a smarter default initial state should be created by the player. For example both videos could automatically be scaled to the same height.



Fig. 6: Tracked new ratios on user resizing the dual player

On the other hand based on the analyses of the new ratios after users resized the dual player most users increase the size of the right video stream that contains the slides. So one solution could be to reinitiate the ratio of the dual player based on the ratio that the user changed on the previous watched video. Another possibility would be to easily allow to toggle to a picture in picture view.

The high amount of videos play in full-screen indicates that users want to optimize the available space while watching these videos. A disadvantage of the full-screen mode is that the context of the page is lost. Therefore for example new in-site notifications are not shown and navigation to the next item requires one interaction more: First leave full-screen, than go to next item).

Concerning the speed only 1 out of 10 video plays happened with a changed speed. As the player currently resets the initial speed on every new video this is lower than expected. Still, it would make sense to store the personal video speed setting, course wide or even per teacher. Over 99% are choosing a higher speed, in our sample data set the average playback speed of those plays was 1.52.

Nal Li et al. show in [7] that events trackable by video analytics data correlate to the difficulty of the video. This includes video speed decreases, frequent and long pauses, infrequent seeks with a high amount of skipping and multiple views by the same user. These findings could be used to automatically determine a difficulty indicator which then could be used to adapt the following self-test and provide extra material. Also if this indicator is significant higher for certain videos within a course this information could be used to inform the teaching team so they can consider recording this video with more explanations or a different didactic approach. Bauer et al [8] had the idea of finding hot spots in recorded university lectures by analyzing the media server's log files to identify scenes where the learners jump back conspicuously often. This can also be used by the teaching team to improve their learning material and didactic approach.

5 CONCLUSION

In this paper the authors showed that there are many variables that MOOC creators have to be aware of when they create and deliver video content in the context of MOOCs. Beside the choice of single stream vs. dual stream there are several other choices, concerning video setup and video length that will impact the learning experience. Allowing offline usage is a heavily requested and used feature. But this often breaks the learning cycle so a full fledged offline learning experience including self tests should be enabled using HTML5 technologies or native apps. Learning Analytics can help to get insights into the detailed media usage to identify weak spots and help the content creators to improve the quality of the provided video materials. Smart video players should offer a rich set interaction. Finally, text content representation of the video is needed to enable functions like search, tag clouds and to provide multi-language support. Several improvement possibilities to increase the Video watching experience in MOOCs have been introduced. These new features have to be implemented and tested in the future.

REFERENCES

- [1] Renz, J., Staubitz, T., Pollak, J., & Meinel, C. (2014). Improving the onboarding user experience in, EDULEARN14 Proceedings
- [2] Malchow, M., Bauer, M., & Meinel, C (2015). Enhance lecture archive search with OCR slide detection and in-memory database technology (submitted)
- [3] Malchow, M., Bauer, M., & Meinel, C. (2015). Lecture Butler - Teaching Reasonable Lectures from a Lecture Video Archive. (submitted)
- [4] Yang, H., Siebert, M., Lühne, P., Sack, H., & Meinel, C (2011). Lecture Video Indexing and Analysis Using Video OCR Technology. In Signal-Image Technology and Internet-Based Systems (SITIS), 2011 Seventh International Conference on, pages 54-61, Nov 2011.
- [5] Renz, J., Staubitz, T., & Meinel, C., (2014) Mooc 2 go, in Proceedings of Mobile Learning Conference, 2014.
- [6] Yousef, A., Chatti, M., Danoyan, N., Thüs, N., & Schroeder, U. (2015), Video-Mapper: A Video Annotation Tool to Support Collaborative Learning in MOOCs, 2015 EMOOCs Proceedings, pages 131-139
- [7] Li, N., Kidzinski, L., Jermann, P & Dillenbourg, P., (2015) How Do in-video Interactions Reflect Perceived Video Difficulty?, 2015 EMOOCs Proceedings, pages 112-120
- [8] Bauer, M., & Meinel, C. (2014). A Concept to Analyze User Navigation Behavior Inside a Recorded Lecture (to Identify Difficult Spots). 13th International Conference on Information Technology Based Higher Education and Training (ITHET).

- [9] Malchow, M., Bauer, M., & Meinel, C. (2014). The Future of Teleteaching in MOOC Times, 17th IEEE International Conference on Computational Science and Engineering (CSE) 2014, pp 438-443