



Open Archive Toulouse Archive Ouverte

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible

This is an author's version published in:

<http://oatao.univ-toulouse.fr/24824>

Official URL

DOI : <https://doi.org/10.1007/s40692-019-00136-6>

To cite this version: Bakkay, Mohamed Chafik and Pizenberg, Matthieu and Carlier, Axel and Balavoine, Eithne and Morin, Géraldine and Charvillat, Vincent *Protocols and Software for Simplified Educational Video Capture and Editing*. (2019) *Journal of Computers in Education*, 6. 257-276. ISSN 2197-9987

Any correspondence concerning this service should be sent to the repository administrator: tech-oatao@listes-diff.inp-toulouse.fr

Protocols and Software for Simplified Educational Video Capture and Editing

Mohamed Chafik BAKKAY · Matthieu HZENBERG · Axel CARLIER · Eithne BALAVOINE · Géraldine MORIN · Vincent CHARVILLAT

Abstract Recently, educational videos have become important parts of e-learning systems which have in turn become widely used due to their flexibility. These videos should be of high quality since higher production values lead to superior learning outcomes. However, creating high quality video is a difficult task for teachers since it needs technical knowledge that includes video recording and timeline usage. Hence, creating educational video production software, that is at the same time easy-to-use and able to produce high quality educational videos, is very advantageous. In this paper, we developed protocols for an easy-to-use piece of software that enables teachers that have little technological background to produce their own educational videos autonomously. In fact, our contribution is to reduce the complexity of the whole video production process by introducing a preparation step based on micro-teaching and upstream specification. An evaluation of the software with six teachers is performed. This evaluation, based on think-aloud protocol and quantitative measurements, showed that the introduction of the preparation step allowed the participant teachers to produce high quality educational videos in less than three hours.

Keywords MOOC · Educational Videos · Educational Specification Standards · Video Acquisition · Video Acquisition · Video Editing

1 Introduction

Over the past few years, a real revolution in e-learning has been happening. It consists in moving lots of content and learning materials online in the form of Massive Open Online Courses (MOOCs) (Li et al. 2018).

University of Toulouse, IRIT, France

One thing that these systems have in common is their focus on delivering content to the learner through educational video sequences. These videos should be attractive and illustrative for students since higher production values lead to superior learning outcomes. However, there was consensus that the video production step is the most expensive component of creating a MOOC, in terms of both time and money (Hansch et al. 2015). Indeed, it is a challenging task for teachers because it needs technical knowledge (video recording, timeline usage...). In fact, there are three educational video production gaps for a teacher. First, there is the generational skills gap consisting in a digital fluency which is divided between students and the post-secondary education faculty. There is also a technological skills gap between the digital skills that educators currently possess and those they should possess. Finally, there is a gap which consists in the existence of a range of digital skill levels between all faculty members, known as skills.

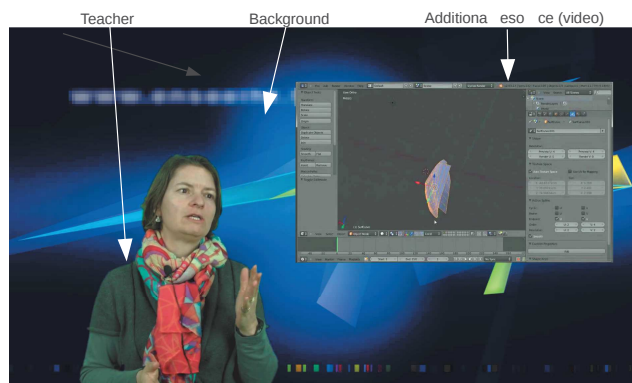


Fig. 1 An example of educational video produced by the proposed software including video-in-video effect.

In fact, the aim of the proposed software is to bridge the above mentioned gaps by simplifying educational video production for teachers (acquisition and editing steps). Our contribution is to introduce a preparation step at the beginning of the video production process. This step brings teachers into their comfort zone, which is the learning environment where they can decompose and organize their lessons. Indeed, this preparation allows teachers that are not familiarized with new technologies to be guided during the acquisition and editing steps and then to be able to create autonomously their own high quality educational videos.

In this paper, we developed software that simplify educational video production in a mini-studio while producing high quality videos. In fact, the proposed software contains three stages. The first stage consists in a preparation step which enables the acquisition of short periods chunks (micro-lessons). This allows to teach content in a most efficient and effective manner possible and to simplify video design, production and update. This step is based on upstream specification for educational video case (green screen acquisition, prompter, video resources, structuring, editing...). The second stage consists of a guided acquisition on a technical environment (a green screen mini-studio). During the acquisition, the teacher, standing in front of the camera, can view her/his comments and annotated resources displayed on a screen. The third stage contains the final video edition step. It is made up of an online web service that has a reduced set of interactive tools using a simplified timeline. A result can be seen in Fig. 1.

The remaining part of this paper is structured as follows: Section 2 is devoted to a brief synthesis of the most relevant works on video production systems. In Section 3, we are going to present the proposed system. An evaluation of the proposed solution is presented in Section 4 in order to demonstrate its effectiveness. Section 5 concludes the paper and presents some directions for future works.

2 Related work

The video production process consists of three phases: planning (pre-production), capture (production) and processing (post-production). It depends on its interaction with users (it can be automatic, semi-automatic, or manual) and its processing time. In this section, we first talk about professional software, their limits and why they should be simplified. Then, we describe how some complexities are reduced by simplifying the software used in both the capture and editing steps. Then, we talk about video production software that are adapted for educational purpose.

2.1 Professional video production software

There are many video editing systems that provide the user with full control over the editing process such as Adobe Premiere (Adobe 2018), Speed Razor (In-Sync 2003) and iMovie (Apple 2018). These professional software offer users a manual video editing tool while including many effects. They allow the creation of high-quality videos. However, they are expensive and complicated to use especially for novices and they need technical knowledge and expertise. In fact, they are based on sophisticated timeline usage which needs special training for a non-professional user. Indeed, they use a frame-based video editing which is very time-intensive since users will operate at a very low level of detail. They are typically targeted at professional users and require extensive training, planning and human coordination. For example, when using professional software tool such as OpenShot (Thomas et al. 2018) to make a video such as shown in Fig. 1, the user has many tasks to do. In fact, she/he needs to place, in order, all recorded videos, images slides, additional resources and the background inside the timeline (Fig. 2). She/he needs also to rescale image slides and additional resources and chooses their coordinates in the screen. Then, she/he needs to choose the chroma key color with a sophisticated configuration of lighting parameters in order to avoid artifacts on foreground and background, etc. These tasks make editing video a difficult, tedious, and error-prone activity for a non-professional user. That is why this type of software should be simplified to allow a novice user such as a teacher to create her/his own educational videos by herself/himself.

2.2 Simplified video production software

Contrary to professional software, simplified software are automatic, easy-to-use and real-time. However, they generally produce low quality videos that are not attractive and illustrative for students. In these software, the simplification could be integrated in different ways. It can be in the acquisition step or in the editing step.

2.2.1 Simplifying video acquisition step

Some works (Uehara et al. 2004; Kumano et al. 2006) proposed an online training-oriented video shooting navigation system to train users how to shoot nice shots for later editing work. (Winkler et al. 2012) presented a solution that uses a depth camera for automatic camera control for tracking a presenter during a talk. (Wulff and Fecke 2012) developed a cost-effective solution for

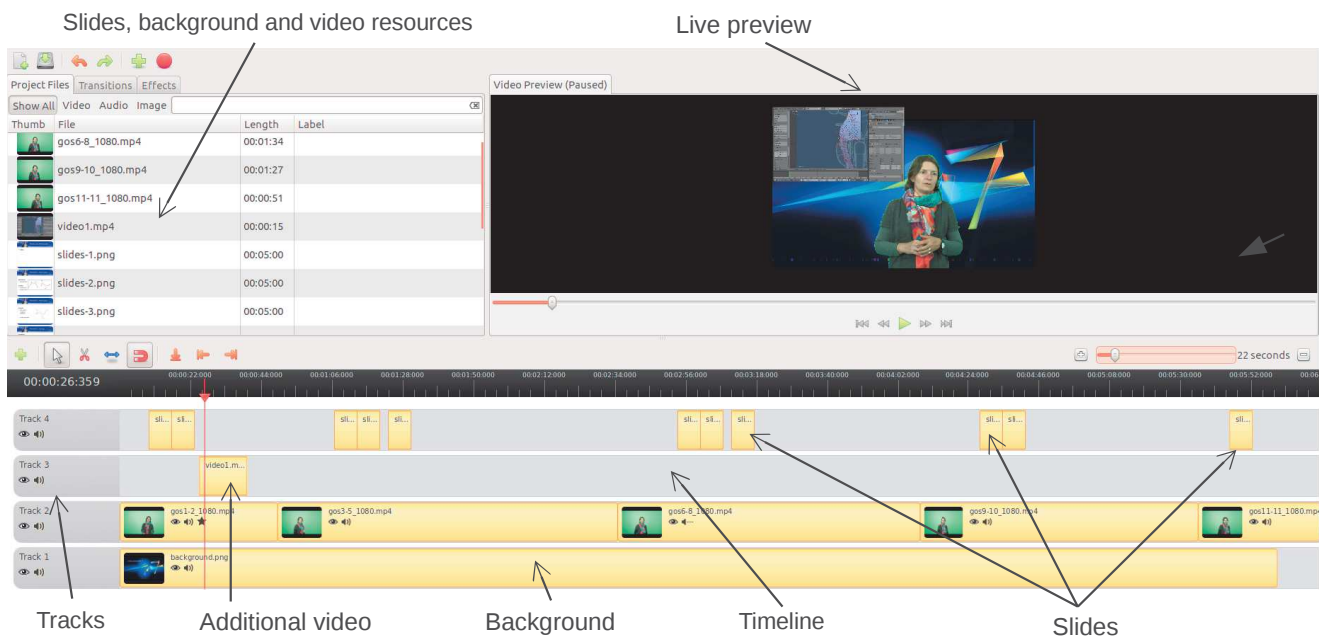


Fig. 2 An example of OpenShot project.

automatic camera control for lecture recordings. (Hulens et al. 2014) proposed an even more advanced system that tracks the lecturer while taking cinematographic rules into account. Some software such as Camstudio (Nickthegeek et al. 2013) record all screen and audio activity on the computer and combine them with web camera stream to create video tutorials for school or college class.

2.2.2 Simplifying video editing step

Some software simplified the editing step. For example, (Shrestha et al. 2010; Bano and Cavallaro 2015) automatically combined the most desirable segments from different recordings resulting in a single video stream. (Casares et al. 2002) used metadata to make digital video editing more accessible to novices. (Lu et al. 2013) simplified the timeline. In fact, they presented a video editing technique based on changing the timelines of individual objects in video in order to leave them in their original places while putting them at different times. LACES (Freeman et al. 2014) is a tablet-based system enabling simple video editing by simplifying interaction with users.

2.3 Educational video production software

To produce educational videos, both acquisition and editing steps should be adapted for pedagogic content (Savov

et al. 2019). Thus, lesson slides, prompter and some additional resources can be included. Indeed, teachers can view, comment on and annotate resources displayed on a screen. Some software focused on the simplification of the acquisition step. For example, T-Cube (Volker et al. 2003) is an easy-to-use multimedia authoring system designed for eLearning. In this software, video, audio and screenshot, are recorded and encoded in the classroom and simultaneously published on the Internet. (Tabuenca et al. 2014) presented a mobile authoring tool for open educational resources to foster universal access to educational resources. (Chen et al. 2015) presented a lecture recording system that employs gestures and digital cameras to facilitate remote-distance teaching. The lecturer can control the pan-tilt-zoom camera (PTZ camera) to record the teaching process by the gestures using Kinect. Computer vision algorithms, like face tracking, are used to offer real-time feedback during such directed actions. Some other methods, used specification standards to simplify the editing step. In fact, there are many specification standards for educational content (SCORM: Sharable Content Object Reference Model, NSL: Narrative Structure Language, ...). These technical standards allow e-Learning platforms to operate together. They allow also instructional designers to distribute their content on a wide variety of e-learning platforms without rewriting efforts. (Zsombori et al. 2011) introduced an approach to the automatic generation of video narratives from user generated content gathered in a shared repository using the stan-

dard NSL. (Sharda 2007) presented a methodology for authoring Educational Multimedia content using story telling as the underlying pedagogical technique. (Wang and Hsu 2006) described a system, which follows the standard SCORM, to separate e-material for use (as teaching templates/learning objects) and to label the material with use of semantic metadata for searching. Finally, the advantage of these educational video production software is that they are adapted for pedagogical content and produce high quality videos. However, they focus either on the simplification of the acquisition step or the simplification of the editing step i.e. they don't simplify the whole video production process.

The research questions of this paper are: “How could we simplify the whole process of educational video production for teachers while producing high quality videos?” (see also (Chew et al. 2018; Storme et al. 2016)) and “Does the preparation step reduce the complexity of educational video production process?” (see also (Zsombori et al. 2011; Sharda 2007)). In fact, the idea of this paper is to reduce the complexity of educational video production by introducing an upstream specification (preparation step) that will result in some constraints related to the educational case (slides, additional video resources, lesson structuring, ...). This preparation step brings teachers into their comfort zone, which is the learning environment where they can decompose and organize their lessons. Indeed, it allows teachers that are not familiarized with new technologies to be guided during the whole video production process. Finally, it allows the teacher to be able to create autonomously their high quality educational videos.

3 Proposed Method

In this paper, we propose an educational video production software that is semi-automatic, simplified and cost-effective. In fact, the main idea of simplification is to introduce a preparation step which is based on micro-learning method (the modularization of learning) to simplify both acquisition and editing steps. This micro-learning method takes place in short byte size unit (micro-lessons) and aims to teach content in a most efficient and effective manner possible. Thus, video content will be easy to design, produce and update in the future. As shown in Fig. 3, the proposed software is composed of three main parts:

- *Preparation step*: making effort here, thinking “micro” and design “granular”
- *Video acquisition step*: shooting video chunks autonomously

- *Video editing step*: video editing as web-service using simplified timeline

3.1 Preparation Step

The teacher, helped possibly by an educational engineer, designs a story, decomposes it in micro-lessons and projects this decomposition on slides. For every slide, she/he can add a text file that will be shown in the prompter. For each group of slides, the teacher also can add some resources (video, audio, image, text). In the example of Fig. 4a, Teacher1 prepared a 5-6 minute video clip broken down into 11 slides and 5 groups of slides. She added video to the first group of slide. The goal of the video chunk was to link the class content to industry standard, in terms of geometric model. The construction of this video chunk was to proceed in five steps:

- Giving the context and illustrating the use of NURBS standard in a industrial software: using additional video
- Recalling the class content and competences of the students: knowledge on polynomial and piecewise polynomial models on one hand, homogeneous coordinates on the other hand;
- Stating the requirements for using the industry standard, and point out its interest (modeling circular shapes)
- Closing the gap between the class content and this requirements: simply express the piecewise polynomial models with homogeneous coordinates leads to rational models, able to model circular shapes;
- Concluding (one slide) to encourage the students to feel competent for the industry standard models.

After preparing micro-lessons, the user should provide to the software a yaml file (specification file) which contains five main sections as described in Fig. 4b:

- *Author section*: contains user information (first name, last name, role).
- *Project section*: contains the title of the video, the slides attribute which represents the path to a pdf file containing the slides to use, the date attribute and the description attribute.
- *Graphical charter section*: provides information used to giving a graphic identity to the video. The logo attribute is the path of the image to use as a logo in the graphic charter. The background attribute stores the path of the image to use as a background in the graphic charter instead of the green screen background.

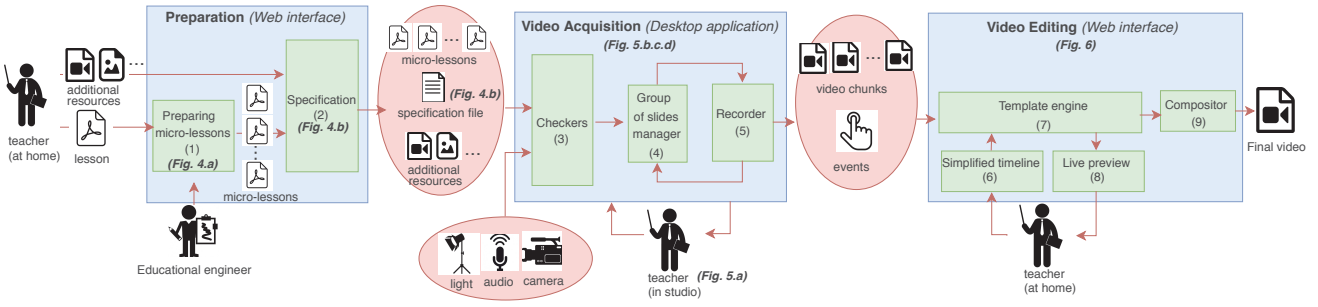
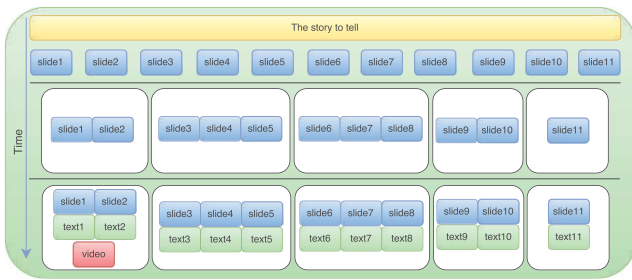
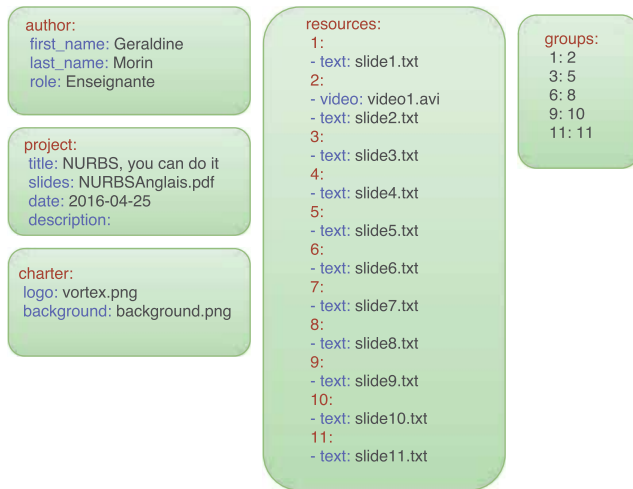


Fig. 3 Flowchart of the proposed software.



(a)



(b)

Fig. 4 Example of preparation step: (a) micro-lessons preparation and (b) structure of the specification file.

- *Resource section:* specifies the additional resources to include in particular slides. The number represents the number of slide on which the resource is included. The attribute represents the type of resource to be used (text or video).
- *Groups section:* specifies the group of slides organization of the presentation. The first number repre-

sents the number of the first slide of the group, and the second number represents the last slide of the group.

3.2 Video Acquisition Step

The proposed software allows a simple way acquisition of video chunks, for teachers. Thus, many process are automated.

3.2.1 Studio

The video is recorded in a green-screen studio. Thus, the green color will be replaced with a chosen background through a chroma keying process. This allows to add features to the video such as the logo of the organization producing the video or some images relevant to the video. As shown in Figure 5 (a) the studio contains:

- *User interface display*
- *Camera*
- *Spotlights*
- *Users controls*
- *Green background*

3.2.2 Checkers

To avoid frustrating the user during acquisition, the proposed software contains a group of checkers, each one focusing on the audio, video, sources and lighting. Thus, if something does not work with the material or the files provided by the user, these checkers prevent the user from recording:

- *Resources checker:* verifies the syntax of specification file. It also verifies that the resources described above are loaded in the correct format and are ready for usage.

- *Video checker*: verifies if the camera is connected to the computer and if the video stream is received through the camera Software Development Kit.
- *Audio checker*: verifies if an audio stream is received. This feature also allows the calibration of the microphone in order to avoid a saturated or too quiet audio channel. When the check is started the program starts recording the user, then playback what have been recorded. Then, the system asks the user if the sound is heard and satisfying.
- *Lighting checker*: verifies the lighting of the studio room, in order to facilitate the substitution of green screen with a chosen background. The camera takes a picture of the user in front of the green screen, then replace that background with a picture. The result is displayed to the user which can improve the lighting of the background by turning on or off the different lights of the studio room.

3.2.3 Group of Slides Manager

The user is able to select the group of slides (described in the specification file) that she/he wants to use for her/his video before recording it. Those groups of slides are displayed through a list with their page numbers below. When a group of slides is selected, the software allows the user to switch between the slides inside this group, while recording the video.

3.2.4 Video Recorder

The video recorder adapts its user interface according to the contents actually being used. It contains a *slider widget*, a *prompter widget*, a *recording widget* and an *additional resource widget* which are displayed according to the specification file. The *slider widget* contains buttons to navigate in the current group of slides. The video recorder stores the time when the switching between slides event is happened in an events yaml file that will be used later in the video editing process. The *prompter widget* contains speed and zoom buttons. The *recording widget* contains controls to start and stop recording and replay. These events are also stored in the events yaml file. After recording, the user can validate her/his video chunk or she/he can record another one and choose the best between the new and the last one. The video recorder compresses the produced videos in a lighter format for video editing purposes.

3.3 Video Editing Step

After recording video chunks, recorded video and generated events are used in the video editing step. In this

step, the user will choose her/his favorite compositing for each video chunk using a simplified timeline. This timeline allows the user to generate the final video while previewing the result in real-time.

3.3.1 Editing Interface

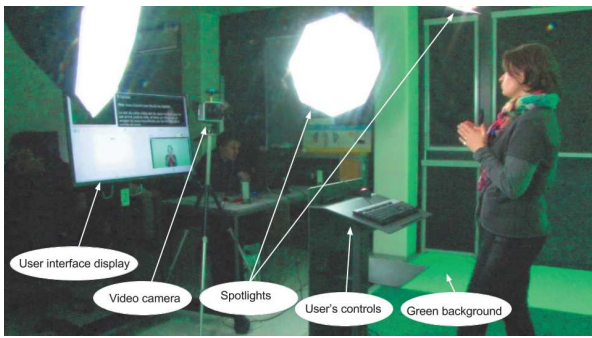
The software provides a user-friendly editing interface using Web development technologies. This interface is organized as on the picture above (Fig. 6). The user is able to select a group of slides using the menu on the right. When she/he chooses the group of slide, the corresponding slides are shown in the editing area. This area contains a preview zone displaying the result of the editing. It contains also a selection of buttons, allowing the user to choose templates. Each template has one particular color. When the user chooses a template for a part of the video, this part appears in the timeline with the color of the chosen template (see the timeline in Fig. 6). The user is able to drag the cursor on the timeline in order to change the current video time. She/he can also click on the timeline areas to change the template type. She/he have also the possibility to drag and drop the timeline delimitation to adjust the video editing. The “Validate” button allows the user to signal that she/he has finished to edit the group of slides. A “Validate all video” button is displayed when all groups of slides have been confirmed. Clicking on this button downloads the video editing file. The proposed software provides also a template recommendation system. In fact, for each selected template, the software recommends the next template that is most suitable (see the red lines in Fig. 6). This helps users to make their videos more dynamic and more attractive for the students.

3.3.2 Template Engine

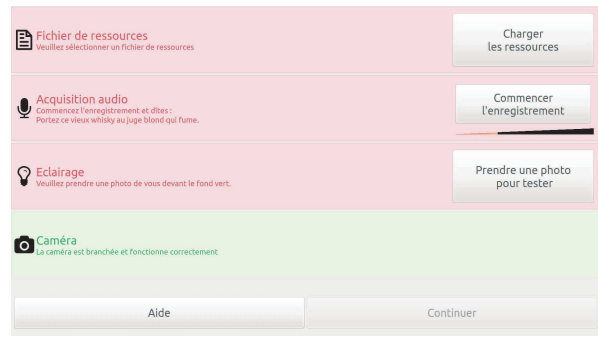
The proposed system provides a template system to simplify the video editing. The proposed system uses EDL (Edit decision list) template files to generate video editing files from the resources characteristics. These files describes the position, the scale, the rotation, the effects to apply (keying for most cases) and opacity of each element. The video editing engine describes the global video editing file using all the files provided by the template engine. This description is done according to the JSON (JavaScript Object Notation) format.

3.3.3 Live Preview

The proposed software provides a web video rendering engine to display a real time preview of the video edit-



(a)



(b)



(c)



(d)

Fig. 5 Video acquisition: (a) green-screen studio (b) checkers interface, (c) group of slides manager interface and (d) recorder interface.

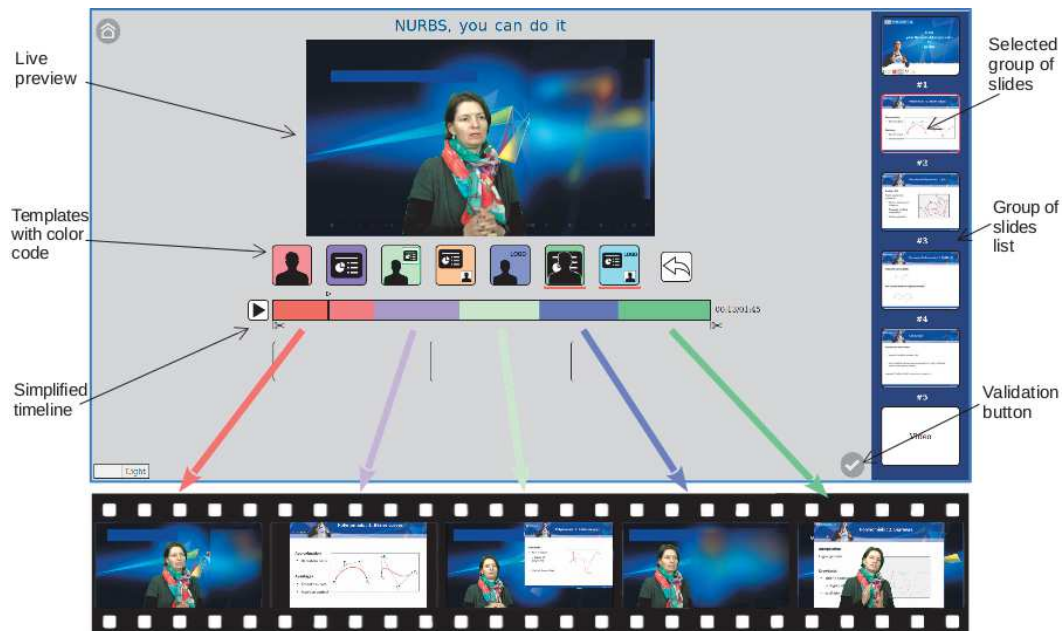


Fig. 6 Editing interface.

ing. This video engine internally uses a timeline representation of the editing step. In fact, it is able to play gaplessly sequences of videos, to combine image and video with a chroma key effect and to combine two

video into one ("picture in picture" effect). It also provide external control for the playback (playing, pausing, muting, seeking at specified time).

3.3.4 The compositor

The compositor uses the files provided by the video editing engine to generate the full resolution video. First, the audio is generated by a Python script that uses FFmpeg library. Then, the video is generated by a C++ program that uses OpenCV and Qt libraries to parse EDL files. Finally, a shell script merges the generated audio and video using FFmpeg library. In fact, the proposed software is organized as a client-server application: it processes the final video on the server side and displays a low-resolution version of the compositing on the client side.

4 Experimental Study

We conducted a user study to evaluate the impact of the preparation step on the simplification of the whole video production process, and the quality of the produced videos. Participants were asked to use the software to perform preparation, acquisition and editing steps. Our participants are six teachers (see Table 1) of different ages and with different levels of familiarity with new technology.

4.1 Procedure

The experiment had three parts: tutorial, experimental tasks, and post-experiment questionnaire. In the first part, participants read the software tutorial. They learned how to use the software. In the second part, each participant performed the experimental tasks. At the end, participants filled out a paper questionnaire about the study in which they were asked about the system in general, the preparation step, acquisition step, the editing step and the tutorial.

During the experimental task, we used both *quantitative evaluation* and *think-aloud protocol*. Think-aloud protocol consists in asking participants to say whatever comes into their minds as they complete the task. This makes thought processes as explicit as possible and then gives us an idea about the participant's cognitive processes.

4.2 Metrics for quantitative evaluation

In the preparation step, we measured:

- *Preparation time*: time needed to prepare the slides and the specification.

In the video acquisition step, we measured:

- *Acquisition time*: time needed for the acquisition;
- *User assistance*: number of interventions of an assistant when user faces a difficulty;
- *Average of capture repetition number*: number of time the user repeat the capture when she/he makes a mistake.

In the video editing step, we measured:

- *Editing time*: time needed for editing;
- *User assistance*: number of interventions of an assistant when user faces a difficulty;
- *Difficulty when using the timeline*: number of clicks on undo button when she/he makes a mistake;
- *Difficulty when choosing templates*: number of clicks on template buttons divided by the number of selected templates; As we mentioned in section 3, the proposed software provides a template recommendation system. Thus, this measurement gives an idea whether the user chooses directly the recommended templates or not.
- *Template usefulness*: number of selected templates divided by number of available templates in the interface. This measurement gives an idea whether the templates available to the user are all useful or not.

4.3 Results and Analysis

- *Think-aloud protocol*: First, most users like the idea of simplifying video production since they were used to having problems with creating their own educational videos: they may spend all the day and the final product was not very satisfying. They affirm also that videos are generally produced by expensive production companies in their universities. We find that nearly all participants made a positive comment about the software.
- *Quantitative evaluation*: We were very encouraged by the fact that all of the study participants were able to succeed in their acquisition and editing tasks, especially given the brevity of the tutorial. In fact, on average, experimental tasks took participants 137.16 minutes. As we can see in Fig. 7a and Fig. 7b, the learning curves for acquisition and editing steps decrease rapidly and they are comparable to the learning curves of simplified software. (We note that the size of chunks are close and vary between 1.1mn and 1.4mn; For example, for teacher1, the size of chunks 1 to 5 are respectively 1.10mn, 1.3mn, 1.40mn, 1.32mn and 1.12mn). At the same time, the quality of the produced videos is comparable to professional videos (Fig. 1).

Table 1 Participant teachers.

	Teacher1	Teacher2	Teacher3	Teacher4	Teacher5	Teacher6
Lesson topic	3D modeling	Machine learning	English	Algebra	Computer network	Image processing
Age	46	48	49	52	48	61
Familiarity with new technology	medium	high	low	medium	high	low
Slides number	11	19	9	75	51	10
Groups of slides number	5	10	4	14	10	6
Prompter usage	yes	no	no	yes	no	yes
Additional resources	yes	yes	no	no	no	no

Table 2 Evaluation of the proposed software.

Participants		Teacher1	Teacher2	Teacher3	Teacher4	Teacher5	Teacher6
Preparation metrics	Preparation time	60min	75min	45min	120min	90min	125min
Acquisition metrics	Acquisition time (mn)	22	34	19	47	39	31
	User assistance	2	1	3	3	1	4
	Average of capture repetition number	1.8	1.6	2	1.2	1.3	2
Video editing metrics	Editing time (mn)	16	19	11	26	21	23
	User assistance	1	1	2	2	1	3
	Difficulty of using timeline	3	2	2	3	2	3
	Difficulty of choosing templates	2	1.5	2.5	2	2.5	2.5
	Templates usefulness	0.5	0.6	0.6	0.1	0.2	0.5
Total Video production time (mn)		98	128	75	193	150	179

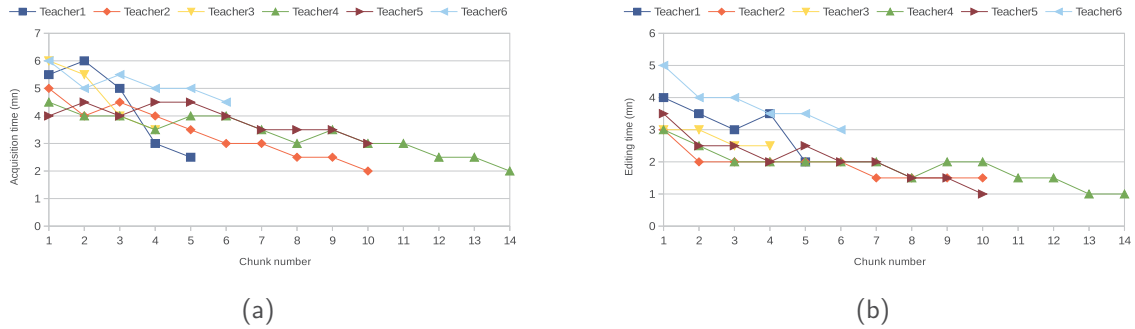


Fig. 7 Learning curves of all participants for (a) acquisition step and (b) editing step.

4.3.1 Preparation

- *Think-aloud protocol*: Users understand quickly the structure of the specification file, they think that the specification file was intuitive. They affirm also that micro-learning aspect of this tool is definitely a plus for any teacher for the organization of their lessons. However, they imagine that maybe a web page and a clearly defined set of slides and text options would help the novices with the extended version of having more degrees of freedom for experts.
- *Quantitative evaluation*: On average, preparation step took participants 85.83 minutes (see Table 2). This means that users spent more than one hour on preparing their lessons before starting the video production. However, the time spent on preparation has an

important impact on reducing the time of acquisition and editing. In fact, this preparation reduces the production time from one day on average to one hour on average.

4.3.2 Acquisition

- *Think-aloud protocol*: Users find that checkers are very useful and have practical ergonomics. They think that the ergonomics of the group of slide selection window and the record interface are practical. They also think that the video production is very user friendly. They affirm that recording micro-lessons separately helps them to avoid errors. However, they suggest to use a pointer to draw inside the slide. They suggest also to use a clicker to change slides

instead of backspace because clicking on the back visually interrupt and distract the user.

- *Quantitative evaluation*: On average, acquisition took participants 32 minutes. The participants repeated their recording at most twice (see Table 2). As we can see in the table, the acquisition time, user assistance and capture repetition number are low for all participants.

4.3.3 Editing

- *Think-aloud protocol*: Users understood quickly the interface. They note that templates are intuitive, easy to understand and numerous enough. However, some users suggest that it would be good to have the ability to move slide transitions. They suggest also to put the voice histogram to know when they can crop the end of the video. Some users suggest the use of a touch screen instead of the mouse.
- *Quantitative evaluation*: On average, preparation step took participants 19.33 minutes. The user assistance, difficulty of using timeline, difficulty of choosing templates scores are low for all participants. However, the template usefulness varies according to the user (Teacher 4 and 5 used only few templates).

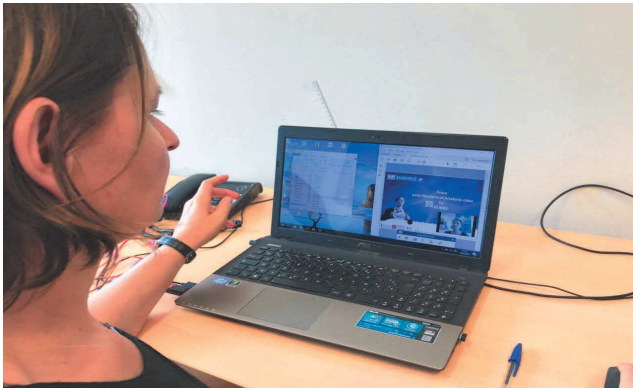


Fig. 8 Evaluation of the video production with Apowersoft (Apowersoft 2017).

4.4 Evaluation of the video production with alternative tools

One year after creating her educational video with our software, we called teacher1 to produce the same video with an alternative tool *Apowersoft* (Apowersoft 2017) for acquisition and *MPEG-streamclip* (Wondershare 2015) for editing. These simplified video production software are bought by our university which puts them at the

disposal of the teachers. First, we showed Teacher1 the tutorials of *Apowersoft* and *MPEG-streamclip*. Then, we asked her to produce two videos: with and without using the preparation step.

4.4.1 Video acquisition without preparation

Teacher1 repeated capture several times. Then, she faced difficulties to exploit the additional video resources in order to make the video-in-video effect. Finally, she asked for her preparation because it simplifies the capture and facilitates the exploitation of the additional videos. Quantitatively, Teacher1 spent 65mn trying to capture and she produced a video with low quality.

4.4.2 Video production with preparation

Thanks to the decomposition in chunk of the preparation step, Teacher1 found the possibility of exploitation of the additional video (video-in-video effect). However, she still has synchronization problem between the recording and additional video starting. She also asked for the possibility of using the prompter. Finally, she produced a video with low quality, but better than the first one. Quantitatively, Teacher1 spend 28mn in acquisition with *Apowersoft*. The learning curve decreases rapidly and is comparable to the learning curve presented in Fig. 7a. She spent 35mn in editing with *MPEG-streamclip*. The learning curve decreases less rapidly than *Apowersoft* curve.

4.4.3 Comparison between our software and alternative tools

Teacher1 affirmed that she preferred using our software for the following reasons. First, although *MPEG-streamclip* and *Apowersoft* were easy to use, they produced low quality videos and they didn't offer a lot of possibilities. Indeed, for acquisition, our software provides chroma keying, picture-in-picture effect and prompter. However, *Apowersoft* provides only screen recorder facilities. For editing, our software provides multiple template, template recommendation system and live preview. However, *MPEG-streamclip* provides only simple timeline editor. Furthermore, the preparation step of our software allowed her to simplify the acquisition step. In fact, each chunk was recorded separately which reduced the possibility of mistakes. Besides our software allowed her to resolve synchronization problem between the recording and additional video starting. Moreover, the acquisition step of our software is more adapted for educational content, and the editing step is simplified using predefined templates. Finally, the quality of the

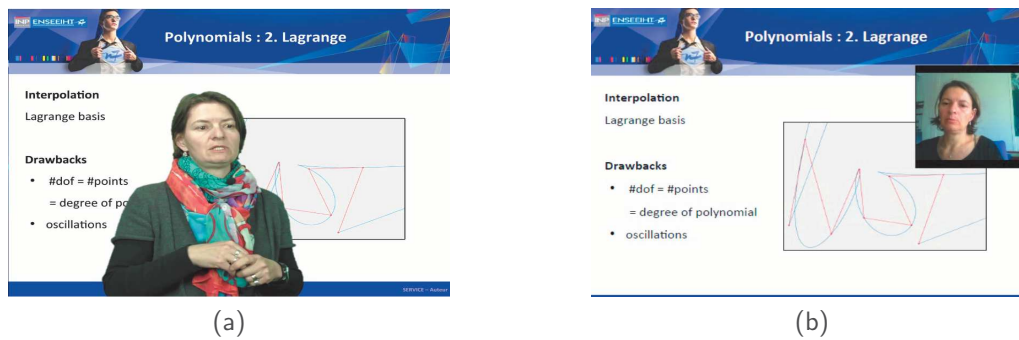


Fig. 9 Videos produced by Teacher1 using (a) the proposed software and (b) Apowersoft.

video produced by our software was higher and comparable with professional video editing software (see Fig. 9).

5 Conclusion

In this paper, we developed an easy-to-use software that enables teachers that have little technological background to produce their high quality educational videos autonomously. Our contribution is the introduction of a preparation step that simplify the whole educational video production process while producing high quality videos. In fact, the preparation step bridges the gaps described in Section I. First, it bridges generational skills gap by bringing teachers into their comfort zone, which is the learning environment where they can decompose and organize their lessons. Indeed, this step bridges also technological skills and inequality skills gaps since it allows teachers that are not familiar with new technologies to be guided during the acquisition and editing steps and then to be able to create autonomously their high quality educational videos. An evaluation of the software with six teachers is performed. This evaluation, based on think-aloud protocol and quantitative measurements, showed that the introduction of the preparation step allowed the participant teachers to produce their videos in less than three hours. It shows also that the learning curves of the acquisition and editing steps decrease rapidly, while the quality of the produced videos is still higher and comparable with the quality of professional video production software. For future works, we will first improve each step of the proposed software, using the feedback of the different users. Second, we will propose an interactive web interface that will help teachers to generate their preparation files. Finally, we will introduce augmented reality techniques to assist users during the acquisition step.

References

- Adobe (2018). Adobe premiere (version 13). <http://www.adobe.com/products/premiere/>.
- Apowersoft (2017). Apowersoft (version 2.9.1). <https://www.apowersoft.fr/>.
- Apple (2018). imovie (version 10.1.9). <http://www.apple.com/imovie/>.
- Bano, S. and Cavallaro, A. (2015). Vicomp: composition of user-generated videos. *Multimedia Tools and Applications*, pages 1–24.
- Casares, J., Long, A. C., Myers, B. A., Bhatnagar, R., Stevens, S. M., Dabbish, L., Yocum, D., and Corbett, A. (2002). Simplifying video editing using metadata. In *Proceedings of the 4th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, DIS '02, pages 157–166, New York, NY, USA. ACM.
- Chen, Y.-Q., Chang, C.-F., and Su, P.-C. (2015). A tabletop lecture recording system based on gesture control. In *Consumer Electronics - Taiwan (ICCE-TW), 2015 IEEE International Conference on*, pages 372–373.
- Chew, S. W., Cheng, I.-L., Chen, N.-S., et al. (2018). Exploring challenges faced by different stakeholders while implementing educational technology in classrooms through expert interviews. *Journal of Computers in Education*, 5(2):175–197.
- Freeman, D., Santosa, S., Chevalier, F., Balakrishnan, R., and Singh, K. (2014). LACES: Live Authoring through Compositing and Editing of Streaming Video. In *ACM CHI Conference on Human Factors in Computing Systems (CHI '14)*, pages 1207–1216, Toronto, Canada.
- Hansch, A., Hillers, L., McConachie, K., Newman, C., Schildhauer, T., and Schmidt, P. (2015). Video and online learning: Critical reflections and findings from the field. *SSRN eLibrary*.
- Hulens, D., Goedeme, T., and Rumes, T. (2014). Autonomous lecture recording with a ptz camera while complying with cinematographic rules. In *Computer and Robot Vision (CRV), 2014 Canadian Conference on*, pages 371–377.
- In-Sync (2003). Speed razor (version 5.5). <http://www.in-sync.com/>.
- Kumano, M., Uehara, K., and Ariki, Y. (2006). Online training-oriented video shooting navigation system based on real-time camerawork evaluation. In *Proceedings of the 2006 IEEE International Conference on Multimedia and Expo, ICME 2006, July 9-12 2006, Toronto, Ontario, Canada*, pages 1281–1284.
- Li, J., Tang, Y., Cao, M., and Hu, X. (2018). The moderating effects of discipline on the relationship between asyn-

- chronous discussion and satisfaction with moocs. *Journal of Computers in Education*, 5(3):279–296.
- Lu, S.-P., Zhang, S.-H., Wei, J., Hu, S.-M., and Martin, R. R. (2013). Timeline editing of objects in video. *IEEE Transactions on Visualization and Computer Graphics*, 19(7):1218–1227.
- Nickthegeek, Windycity, and Wolfpackmars2 (2013). Camstudio (version 2.7). <http://camstudio.org/>.
- Savov, S. A., Antonova, R., and Spassov, K. (2019). Multimedia applications in education. In *Smart Technologies and Innovation for a Sustainable Future*, pages 263–271. Springer.
- Sharda, N. K. (2007). Authoring educational multimedia content using learning styles and story telling principles. In *Proceedings of the International Workshop on Educational Multimedia and Multimedia Education*, Emme '07, pages 93–102, New York, NY, USA. ACM.
- Shrestha, P., de With, P. H., Weda, H., Barbieri, M., and Aarts, E. H. (2010). Automatic mashup generation from multiple-camera concert recordings. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, pages 541–550, New York, NY, USA. ACM.
- Storme, T., Vansielegheem, N., Devleminck, S., Masschelein, J., and Simons, M. (2016). The emerging pedagogy of moocs, the educational design of technology and practices of study. *Journal of Computers in Education*, 3(3):309–328.
- Tabuenca, B., Kalz, M., Ternier, S., and Specht, M. (2014). Mobile authoring of open educational resources for authentic learning scenarios. *Universal Access in the Information Society*, pages 1–15.
- Thomas, J., Finch, A., McCall, H., Girard, O., Karlinux, T.J., and Lavault, M. (2018). OpenShot (version 2.4.2). <http://www.openshot.org/>.
- Uehara, K., Amano, M., Arika, Y., and Kumano, M. (2004). Video shooting navigation system by real-time useful shot discrimination based on video grammar. In *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on*, volume 1, pages 583–586 Vol.1.
- Volker, M. M., Schillings, V., Chen, T., and Meinel, C. (2003). T-cube: A multimedia authoring system for elearning. In *Proceedings of E-Learn*, pages 2289–2296.
- Wang, H.-C. and Hsu, C.-W. (2006). Teaching-material design center: An ontology-based system for customizing reusable e-materials. *Computers & Education*, 46(4):458–470.
- Winkler, M. B., Höver, K. M., Hadjakos, A., and Mühlhäuser, M. (2012). Automatic camera control for tracking a presenter during a talk. In *ISM*, pages 471–476. IEEE Computer Society.
- Wondershare (2015). Mpeg-streamclip (version 1.2.1). <http://www.mpegstreamclipmac.com/>.
- Wulff, B. and Fecke, A. (2012). Lecturesight - an open source system for automatic camera control in lecture recordings. In *Multimedia (ISM), 2012 IEEE International Symposium on*, pages 461–466.
- Zsombori, V., Frantzis, M., Guimaraes, R. L., Ursu, M. F., Cesar, P., Kegel, I., Craigie, R., and Bulterman, D. C. (2011). Automatic generation of video narratives from shared ugc. In *Proceedings of the 22Nd ACM Conference on Hypertext and Hypermedia*, HT '11, pages 325–334, New York, NY, USA. ACM.