5-2024

SwapVid: Integrating video viewing and document exploration with direct manipulation

Taichi MURAKAMI

Kazuyuki FUJITA

Kotaro HARA
Singapore Management University, kotarohara@smu.edu.sg

Kazuki TAKASHIMA

Yoshifumi KITAMURA

Citation

MURAKAMI, Taichi; FUJITA, Kazuyuki; HARA, Kotaro; TAKASHIMA, Kazuki; and KITAMURA, Yoshifumi. SwapVid: Integrating video viewing and document exploration with direct manipulation. (2024). *CHI '24: Proceedings of the CHI Conference on Human Factors in Computing Systems, Honolulu, May 11-16*. 1-13.
Available at: https://ink.library.smu.edu.sg/sis_research/8921

SwapVid: Integrating Video Viewing and Document Exploration with Direct Manipulation

Taichi Murakami
Tohoku University
Sendai, Japan
taichi.murakami.s5@dc.tohoku.ac.jp

Kazuyuki Fujita
Tohoku University
Sendai, Japan
k-fujita@riec.tohoku.ac.jp

Kotaro Hara
Singapore Management University
Singapore
kotarohara@smu.edu.sg

Kazuki Takashima
Tohoku University
Sendai, Japan
takashima@riec.tohoku.ac.jp

Yoshifumi Kitamura
Tohoku University
Sendai, Japan
kitamura@riec.tohoku.ac.jp

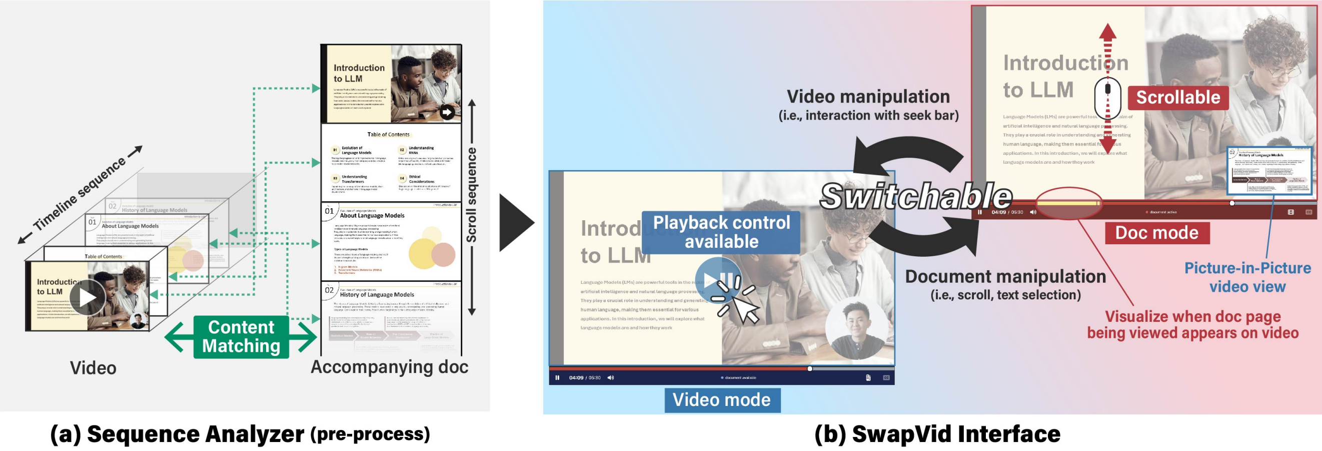

Timeline sequence
Scroll sequence
Content Matching
Video
Accompanying doc
Introduction to LLM
Table of Contents
About Language Models
History of Language Models
Video manipulation
(i.e., interaction with seek bar)
Switchable
Document manipulation
(i.e., scroll, text selection)
Scrollable
Playback control available
Doc mode
Video mode
Picture-in-Picture video view
Visualize when doc page being viewed appears on video
(a) Sequence Analyzer (pre-process)
(b) SwapVid Interface

Figure 1: SwapVid is a novel interface that supports document-based video viewing; (a) our *sequence analyzer* performs content matching between each video frame and the document; (b) based on this, the interface integrates the video viewer and document viewer into a single window and seamlessly switches between them upon the user's direct manipulation of the video/document.

ABSTRACT

Videos accompanied by documents—*document-based videos*—enable presenters to share contents beyond videos and audience to use them for detailed content comprehension. However, concurrently exploring multiple channels of information could be taxing. We propose SwapVid, a novel interface for viewing and exploring document-based videos. SwapVid seamlessly integrates a video and a document into a single view and lets the content behaves as both video and a document; it adaptively switches a document-based video to act as a video or a document upon direct manipulation (*e.g.*, scrolling the document, manipulating the video timeline). We conducted a user study with twenty participants, comparing SwapVid to a side-by-side video/document views. Results showed that our interface reduces time and physical workload when exploring slide-based documents based on video referencing. Based on the study findings, we extended SwapVid with additional functionalities and demonstrated that it further extends the practical capabilities.

CHI '24, May 11–16, 2024, Honolulu, HI, USA
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0330-0/24/05
https://doi.org/10.1145/3613904.3642515

CCS CONCEPTS

• **Human-centered computing** → **Interactive systems and tools**; *User interface design*; *Open source software*; *User studies*; Web-based interaction.

KEYWORDS

Document-based video interfaces, Video-document matching, Lecture videos, Screen-shared documents

ACM Reference Format:
Taichi Murakami, Kazuyuki Fujita, Kotaro Hara, Kazuki Takashima, and Yoshifumi Kitamura. 2024. SwapVid: Integrating Video Viewing and Document Exploration with Direct Manipulation. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24), May 11–16, 2024, Honolulu, HI, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3613904.3642515

1 INTRODUCTION

Videos accompanied by PDF documents, slides, or program scripts that are tightly coupled in contents—what we call *document-based videos*—enable presenters in video lectures, meetings, and conference presentations to share more contents than what they can share through a video channel alone. In a Zoom[1] meeting or a Coursera[2] lecture, for example, a presenter screen shares the document, shows the part of the document that s/he is referring to, and talks over the content, helping ground the video lecture/conversation topic with the audience. The use of document-based videos has rapidly increased, particularly due to the recent COVID-19 pandemic that forced most communication to happen online video calls. While document-based videos benefit presenters by giving them more channels to deliver content to their audiences, audiences tend to get more cognitively taxed because they must navigate between what is discussed on a video and the contents of the documents. For example, when viewing a screen-shared slide in Zoom, the audience needs to deliberately navigate to the corresponding section of the accompanying document in a separate window to view the pages around what is being screen-shared. Although many studies have been conducted to support user comprehension and exploration of lecture and presentation videos [39, 49, 55, 57, 60], the difficulty of such navigation tasks has not yet been addressed.

To design a novel tool that addresses difficulties in attending to both a video and the accompanying document, we draw inspiration from the prior work that introduced direct manipulation in navigating videos [13, 31]. Dragicevic et al. [13] created a video player that allows users to control the playback position of a video by directly interacting with video content (*e.g.,* by dragging a ball in a billiard video); this was achieved by extracting motion data from the video. Denoue et al. applied a similar idea of direct manipulation in screen-based tutorial videos and developed an early prototype that allowed a video viewer to scroll and zoom a document within a video, as if they were real documents [12]. This interface inspired us with its potential to facilitate navigation between videos and documents on a single viewer. However, their tool is limited by its implementation which relied on video data alone; it only allowed the user to interact with a part of the document that was presented in the video. We believe that this idea of directly manipulating a document on a video screen could be improved by better blending video and document data when accompanying documents are available (rather than processing a potentially fragmented document presented on a video).

In this paper, we propose a novel interface called SwapVid. SwapVid superimposes a video viewer and a document viewer on top of each other, and adaptively brings one viewer on top of another. Since this adaptive switching of viewers is done seamlessly upon user's direct manipulation on the viewer (Fig 1b), our interface supports user navigation between video and documents in a single window. That is, when the user scrolls the content within the video, our interface switches the mode to a document viewer with almost no visual change, thereby supporting the user's transition from video to document and scroll through the accompanying document in the same view. Conversely, when exploring a document, our interface highlights segments on the video timeline where the content of the document being viewed appears in the video (Fig 1b right), supporting the user's seamless transition from document to video upon direct manipulation of the timeline. Such interactions are enabled by visually synchronizing each video frame with the corresponding part of the document using our OCR-based video-document matching algorithm (Fig 1a). This study explores the SwapVid prototype for viewing prerecorded video, although we believe the idea can also be applied to real-time video viewing such as Zoom video calls, except that the video timeline cannot be scrolled into the future.

To examine SwapVid's effectiveness in supporting the user's smooth transition between the video and the documents and its overall usability, we conducted a user study with twenty participants, comparing SwapVid to a conventional side-by-side video/document views in a video summary task and two types of content exploration tasks. Results show that our interface helps the user reduce physical and cognitive workload, especially in document exploration tasks based on video referencing. Results also show that our interface is preferred over the conventional interface in the content exploration tasks. Based on these results, we conclude that our interface successfully facilitates the user's concurrent exploration between a video and a document.

The main contributions of this paper are as follows:

- Design and implementation of a SwapVid prototype, enabled by an algorithm that estimates the viewport of the document that appears in the video and an interface that adaptively switches video/document views upon the user's direct manipulation,
- A user study (N=20) demonstrating that SwapVid reduces the time and physical workload when exploring slide-based documents based on video referencing, and
- The code for SwapVid that works with pre-existing document-based videos as a web application with enhanced practical capabilities as described in Section 6.

2 RELATED WORK

This section summarizes previous efforts on video interfaces and real-time document-sharing interfaces.

2.1 Seek-bar-based Video Interfaces

Most common video players on the web today, such as YouTube[3], use a seek-bar-based interface. A typical problem with the seek bar is that its operation is somewhat demanding, especially for long video content. To solve this, many studies have considered various improved interfaces with variable granularity of seeking [26–28, 38, 52]; examples include ZoomSlider [27], which allows the user to change the scale of the timeline slider by moving the cursor vertically. Another approach, more relevant to this study, is to facilitate user navigation with semantic visualization. Many interfaces have attempted to provide annotation on the timeline based on certain content processing and/or using video metadata [1, 6, 8, 17, 32, 54, 59]. As a practical example, YouTube also has the feature to visualize the most replayed parts on a timeline to

[1] https://zoom.us/

[2] https://www.coursera.org/

[3] https://www.youtube.com/

help user navigation. Based on the usefulness observed in these instances, we also consider using annotations on the seek bar to indicate the correspondences between a video and its accompanying document in our interface.

2.2 Content-based Video Interfaces

Some prior work have sought to design storyboard-like interfaces, which allow users to navigate video by manipulating an interface specialized by the video content [22, 29, 30, 48, 51]. For videos of lectures or presentations, research has introduced a variety of storyboard-like interfaces to efficiently navigate lengthy videos, such as one with text digests and thumbnails for each segment of the video [49], crowdsourced concept maps of the talks [41], and control panels with semantic cues based on multimedia analysis within the video [60]. These efforts would enrich the viewing experience of document-based video, but they do not specifically address the tasks of content navigation between video and documents.

Other attempts with a more similar motivation to ours are to synchronize video timelines with related visual entities, to reduce user workload in controlling playback and/or switching focus between content [9, 14, 15, 36, 39, 50, 55]. As for examples targeting slide-based presentation videos, several studies have explored ways to synchronize presentation videos with slide pages by analyzing them using OCR [55] or image feature matching [14, 15]. These efforts are certainly a key component for indexing and better browsing of the videos. However, they focus mainly on the video-processing algorithms, not on how to effectively navigate between the video and the document from a user interface perspective. As one of the few research efforts mentioning the user interface, Li et al. developed an interface that synchronizes scrolling of the textbook with the progress of lecture videos [39]; yet, it only arranges the video and textbook views side by side, which would be limited by the large amount of screen space required. We believe that navigation between videos and documents could be made much smoother, without splitting the view, by better blending the two interfaces.

2.3 Direct Video Manipulation Interfaces

Another notable approach to video interfaces is direct manipulation [13, 31]. The early idea is to detect moving objects in the video and allow the user to move them directly (*e.g.,* by dragging), thus allowing intuitive forward/rewind operations of the video frames. Such interaction paradigm has been applied to various use cases for video manipulation [11, 12, 33, 37, 44–46]. Notably, Denoue et al. developed an interface that allows direct scrolling and zooming of screen-based tutorial videos as if the user was manipulating the actual documents [12], enabled by their own technique to extract document elements from the video frames in real time [10]. This interface inspired us with its potential to support navigation between videos and documents without occupying additional screen space. However, their interface is limited by its implementation relying on video data alone; with their prototype, documents can only be viewed within the ranges that previously appeared in the video, and the image quality of the documents depends on that of the video data. Therefore, we consider significantly extending their interface [12] based on the precondition that accompanying documents are available.

2.4 Real-time Document-sharing Interfaces

Screen sharing is a way to quickly share documents between users in real time [18, 19]. Today, most video call software such as Zoom, Google Meet[4], Skype[5], and Microsoft Teams[6] all include screen-sharing functionalities and have been extensively used for online communication involving documents. Research on video call interfaces has considerably increased recently, some of which are related to screen sharing. For example, some interfaces allow each of the video call participants to recognize which part of the document their verbal [43] or nonverbal actions such as pointing [21] are referring to. However, their motivation is somewhat different from us; we focus on the limitation that screen sharing is inherently a WYSIWIS (what-you-see-is-what-I-see) interface, which does not allow each user to directly manipulate the displayed video feeds.

To overcome the limitation of WYSIWIS interfaces, research has explored interfaces that allow users to flexibly loose coupling of content between users (*e.g.,* [5, 20, 53]). As an example of interfaces currently in use, Microsoft Teams offers the ability to share slide data itself instead of screen-shared video feeds, allowing each user to navigate through the document independently when page position synchronization with the presenter is turned off. This functionality, while limited to a slide file format (pptx), would be a promising solution to streamline navigation between the document and the presenter's view for real-time document-sharing. Inspired by this idea, we consider our interface to allow the user to manipulate the document itself instead of the video as needed. In addition, we try to apply this idea of real-time document-sharing to general document-based video interfaces, including prerecorded videos, by developing a content matching algorithm between videos and accompanying documents.

3 SWAPVID

We propose SwapVid, a novel interface for viewing and exploring document-based videos to better support content navigation between a video and an accompanying document. The interface integrates a video viewer and a document viewer into a single window and can seamlessly switch between them with direct user manipulation of the video/document. Such interface design is enabled by our algorithm that estimates the viewport (*i.e.,* scroll position and zoom level) of the document that appears in each frame of the video. The SwapVid prototype consists of two components: a ***sequence analyzer*** for pre-generating a mapping between the video timeline and the document location (Figure 1a), and an ***integrated user interface*** with an overlaid video viewer and document viewer (Figure 1b). The remainder of this section describes the design considerations of our interface and the prototype implementation that works with on-demand videos (*i.e.,* prerecorded videos).

3.1 Design Considerations

Our review of prior works highlights that few studies have addressed how interfaces deal with the problem of user workload with two concurrent channels of information. Here, to better frame

[4] https://meet.google.com/
[5] https://www.skype.com/
[6] https://teams.microsoft.com/

the problem, we define the following two tasks that require content exploration between videos and documents, while considering typical scenarios of viewing lecture videos involving documents reported in the previous work [23, 34, 35]:

- **Video-based document exploration task** (**V2D task**, in short) refers to the task of searching an accompanying document for specific information related to the video content being viewed. Typical examples include referring to the previous/next page of a screen-shared document in an online meeting such as Zoom, or returning to the page of a slide that defines a technical term when it is mentioned again in the lecture video.
- **Document-based video exploration task** (**D2V task**, in short) refers to the task of searching a video for information related to the corresponding document being viewed. A typical example would be to find the time period(s) where a certain page of the document being viewed is mentioned in the video.

To support the above two tasks, we consider integrating a video and a document into a single view and seamlessly swapping between them with direct user manipulation. More specifically, the video-to-document transition is triggered by document manipulation (*e.g.,* scrolling the view) and the document-to-video transition is triggered by video manipulation (*e.g.,* interacting with the timeline). Since the interface uses such common actions necessary for content exploration as the triggers to swap between modes, the user can naturally switch between them without learning additional operations.

In addition, to maintain context when switching between video and document, the interface needs to be aware of the relationship between the video timeline and the displayed content locations in the document. To achieve this, we consider creating a video-document matching algorithm to detect which part of the document is shown at each frame of the video. With this algorithm, we additionally introduce a visualization function that highlights on the timeline when the part of the document being viewed is shown in the video. This function will further support D2V tasks.

Based on the above design considerations, we implemented an initial prototype of SwapVid. Although the concept of our interface is applied to both on-demand and real-time video viewing, this early prototype is for on-demand video, with slides and articles as the assumed document formats. Note, we determine the document type solely by its aspect ratio on our prototype: portrait documents are considered articles and landscape documents are considered slides. The prototype is designed to be used with a mouse on a PC, which is a common style for viewing document-based videos. The implementation uses HTML5, CSS and React, a JavaScript UI library, and runs as a web application.

3.2 Sequence Analyzer

3.2.1 Overview. To estimate the viewport of the document that appears in the video, we employed OCR-based content matching, similar to the previous work that attempted content extraction from lecture videos [57] and live presentation videos [55]. We also tried content matching based on image features (*e.g.,* AKAZE [3]), but found them impractical in terms of computational cost and accuracy. We used OCR for the documents as well as the video because it was more convenient to match the position of the bounding box containing the text between the two data.

Figure 2 shows the overall workflow of the sequence analyzer. On one hand, the system performs OCR processing (using Tesseract OCR[7]) on the entire PDF document and creates index data of the contained text segments, consisting of the extracted text strings and their location (*i.e.,* the page number and the coordinates of the bounding box within the page). On the other hand, the system also performs OCR on the video image at every keyframe and extracts the text strings with their locations. The keyframes are obtained when a scene change is detected in the video during its monitoring at $1Hz$ by calculating the differences of the horizontal and vertical projection profile between frames. The viewport estimation process by matching the extracted text segments between the video and the document consists of scroll position estimation and zoom level estimation, as described below.

3.2.2 Scroll position estimation. The system performs a brute-force search to check if each segment of the text strings extracted from the video keyframes (with a string length of ten or more) matches any of those in the index data of the document. Considering that the text segmentation results may be imprecise, match detection is based on exceeding predetermined threshold values for each of the following three similarity indices: string length similarity, N-gram ($N = 2$) similarity, and string sequence similarity.

Since there is usually a difference in the scrolling continuity between slide- and article-type documents (*i.e.,* articles are often scrolled continuously, while slides are shown page by page on the video), the scrolling position estimation process differs slightly depending on the document type. For slides, if the system detects a match in a certain text segment between the video and the document, it returns the page number of the document containing the string as an estimation candidate. After repeating this process from the top of the image at the keyframe, the system finally outputs the most likely page number as the estimated scroll position. For articles, the system uses a certain number (three, in our prototype) of consecutive lines of extracted strings in a video frame as the unit of the match detection. If a match is detected, the system returns the line position containing the string information as a candidate of the estimated scroll position, and repeats this process to output a final estimation result.

3.2.3 Zoom level estimation. Document-based videos may contain scenes that show zoomed-in/out views of a document, so the system is required to estimate the zoom level. To achieve this, our implementation focuses on the size difference of a certain text segment matched between the video and the document. Specifically, the system estimates the zoom level by calculating the ratio of the bounding-box height of the matched string between both data. We used height (not width) because height was observed to be more robust in OCR than width for documents with horizontal texts. The estimated viewport is finally calculated by combining the estimated scroll position and zoom level with the exact size of the video/document.

[7] https://github.com/tesseract-ocr/tesseract

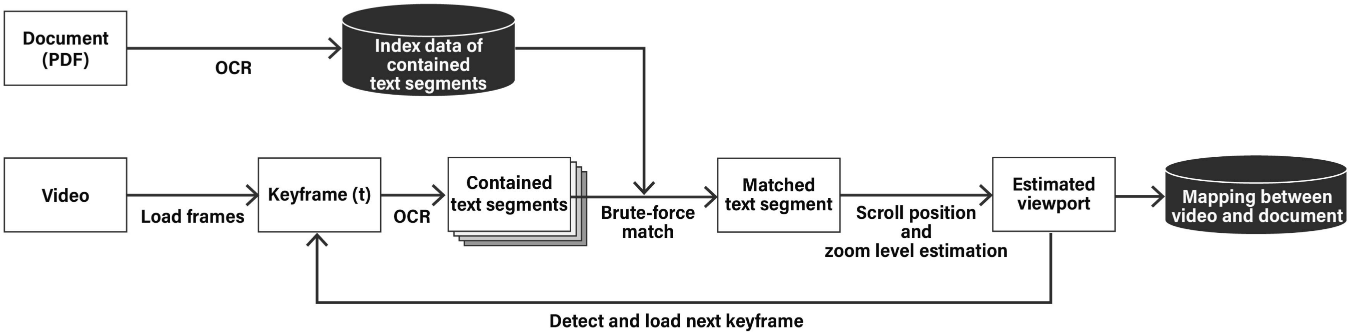

Document (PDF)
OCR
Index data of contained text segments
Video
Load frames
Keyframe (t)
OCR
Contained text segments
Brute-force match
Matched text segment
Scroll position and zoom level estimation
Estimated viewport
Mapping between video and document
Detect and load next keyframe

Figure 2: Workflow of sequence analyzer.

Table 1: Video/document materials used in this study and results of performance test of our sequence analyzer.

Material ID	Doc info		Video info		Results of performance test		
	Source	# of pages	Source	Length	# of keyframes	Correct estimation ratio	Process time per frame
Slide01	[16]	9	YouTube[8]	5:04	14	100%	0.031s (SD = 0.11)
Slide02	[24]	12	YouTube[9]	6:41	14	85.7%	0.028s (SD = 0.089)
Slide03	[47]	12	YouTube[10]	6:54	8	100%	0.023s (SD = 0.087)
Article01	[40]	10 (edited)	YouTube[11]	5:15 (edited)	9	78.5%	0.12s (SD = 0.29)
Article02	[4]	10 (edited)	YouTube[12]	5:29 (edited)	14	100%	0.10s (SD = 0.39)
Article03	[25]	4 (edited)[13]	YouTube[14]	5:53 (edited)	16	87.5%	0.26s (SD = 0.63)

3.2.4 Performance test of sequence analyzer. We tested the accuracy and process time for the viewport estimation of our sequence analyzer. We used six existing document-based videos (three slides and three articles, Slide01-03 and Articles01-03 in Table 1) for the test, all of which are available on the web. These videos did not include zoomed-in/out views. Of the 193 keyframes extracted by the system in these videos, we used 75 frames for the test by manually eliminating irrelevant frames (*e.g.,* those displaying non-documents or playing full-screen slide-embedded videos) and inappropriate frames (*e.g.,* those unable to read the content while scrolling).

We calculated *correct estimation ratio* of the viewport for all targeted keyframes. We defined the "correct" estimation as the system's estimated viewport position with an error of 50 px (equivalent to about 5% of the height of the video view) or less from the ground truth. Since there was no ground truth data for the viewport position in each video, we created them by manually aligning the document position to the video at each keyframe.

As Table 1 shows, the mean correct estimation ratio was 94.4% for slides and 88.7% for articles. This result shows that the system was generally capable of tracking the correct viewport of the documents. Regarding Article01, the reason for the relatively lower accuracy is because the document version that is used for the video differs slightly from the publicly-available one in some sentences.

The mean process time to estimate the viewport for each video keyframe was 0.027s for slides and 0.16s for articles. Given that this process is performed less frequently than $1Hz$, these results would be sufficiently feasible for using our interface on common PCs even with real-time video.

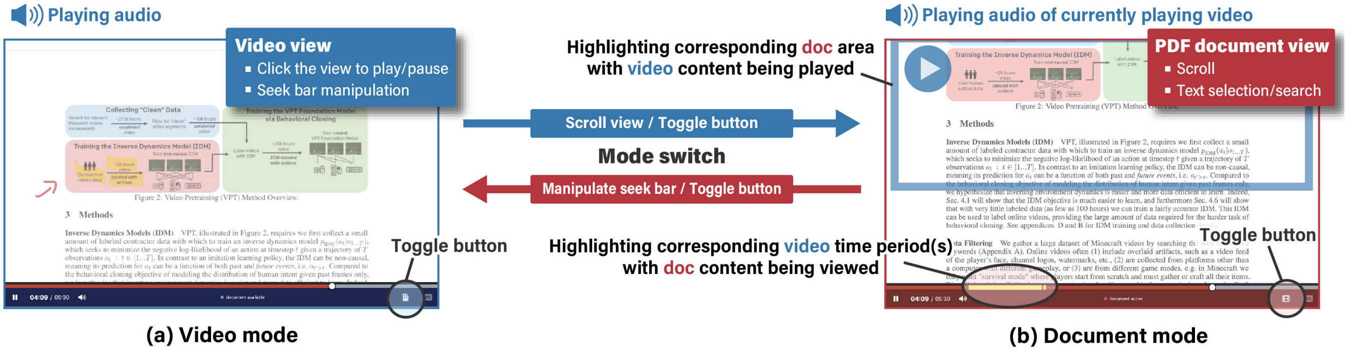
3.3 Integrated User Interface

3.3.1 Overview. Figure 3 shows an overview of the interface provided by SwapVid. Our integrated interface is presented in a single window and switches between video mode (Figure 3a) and document mode (Figure 3b). Switching modes is done by scrolling the view (from video to document) and by manipulating the seek bar (from document to video). The interface also provides a toggle button at the bottom of the screen to explicitly switch the mode, which is particularly useful when the sequence analyzer cannot find a mapping because the video being played is not included in the document. The transition between the modes is implemented by toggling the show/hide of the document viewer, which is superimposed on the video viewer. In the following, we describe the specific interaction and behavior when switching the viewing mode.

3.3.2 Transition from video mode to document mode. When in the video mode, the system switches to the document mode when it detects a scrolling operation by the user. Since the viewport of the document is kept in sync with that of the video, visual changes during mode switching are minimized, allowing the user to experience it as if the document was just scrolled.

The mode transition from video to document is possible either while the video is playing or paused. If transitioned while a video is playing, the video continues playing so that the user can continue to listen to its audio. This means the user can manipulate the document asynchronously from the video, which is an essentially different feature from video direct manipulation interfaces (*e.g.,* [12]). This feature will allow users to use this interface in real-time video situations (*e.g.,* screen sharing via Zoom); the user can look at

[8] https://www.youtube.com/watch?v=EuBmw3hVEuM
[9] https://www.youtube.com/watch?v=U82chCRQVwo&t=1050s
[10] https://www.youtube.com/watch?v=U82chCRQVwo&t=580s
[11] https://www.youtube.com/watch?v=t3Yh56efKGI
[12] https://www.youtube.com/watch?v=ODat7kfZ-5k
[13] The video presents a web page instead of a PDF document, so we used screenshots of this as the accompanying document.
[14] https://www.youtube.com/watch?v=jzdodD3mX_o


Playing audio
Video view
Click the view to play/pause
Seek bar manipulation
Toggle button
Highlighting corresponding doc area
with video content being played
Scroll view / Toggle button
Mode switch
Manipulate seek bar / Toggle button
Highlighting corresponding video time period(s)
with doc content being viewed
Playing audio of currently playing video
PDF document view
Scroll
Text selection/search
Toggle button

(a) Video mode

(b) Document mode

Figure 3: Overview of SwapVid's interfaces in each mode.

other pages in the document without missing what the presenter is saying.

3.3.3 Transition from document mode to video mode. When in document mode, the system switches to the video mode when it detects interaction with the seek bar by the user. While the user is exploring the document, the system keeps monitoring if the currently displayed content exists in the video, based on the mapping generated by the sequence analyzer. If a match is detected, the system visualizes the corresponding time period(s) on the seek bar by highlighting them in yellow (as shown in Figure 3b). Even in the case of real-time video viewing, a seek bar could be placed like in YouTube Live to support video navigation into the past. In addition, when a part of the document view matches the content with the currently playing video, the matched area is visualized in the document viewer as a blue rectangle (as shown in Figure 3b). These two types of visualizations support D2V tasks.

3.3.4 Interface of each viewer. The video viewer provides basic playback functions like many common video players on the web (*e.g.,* YouTube), such as playback, pause, seek bar control (with a preview when the cursor hover over the seek bar), and subtitle display. We implemented them using HTML5, CSS and React (JavaScript).

The document viewer acts as a basic PDF viewer, supporting interactions such as scrolling, text selection, and text search. We used React-pdf[15], a React wrapper for PDF.js, to directly load PDF document data.

4 USER STUDY

We conducted a user study to evaluate SwapVid compared with a conventional interface. The main objectives of the user study were (1) to investigate the performance of navigating contents between a video and an accompanying document using our interface, and (2) to understand the overall usability of the interface. The results of the study showed that SwapVid helps the user reduce time and physical workload in navigating slide-based documents in exploration tasks.

4.1 Participants

Twenty university students (12 males and 8 females, mean age= 23.2 ± 1.64) participated in this experiment. Their university and department affiliations were diverse, but all of them had experience viewing document-based videos for their courses and were familiar with existing interfaces for video viewing (*e.g.,* YouTube) and document exploration (*e.g.,* Adobe Acrobat Reader).

4.2 Experimental Design

The experiment was a two-factor within-subjects design with interface (SwapVid, baseline) and document type (slide, article) as the factors. For the baseline interface, we used a side-by-side separate view, where the participant could control the video timeline and the scroll position of the document independently, a setup that people commonly employ when interacting with a video and a document concurrently (Fig 4a left). To explore the effect of document types with different aspect ratios and text densities on exploration performance, we included document type (*i.e.,* slides and articles) as another factor.

We designed three tasks: (1) summary task, (2) video-based document exploration (V2D) task, and (3) document-based video exploration (D2V) task. The summary task was to provide a summary of a given video while using each interface. We designed this task to simulate real-time video viewing and aimed to examine the overall usability of each interface through actual use. In contrast, we designed the two exploration tasks (*i.e.,* V2D task and D2V task) to assess the SwapVid's efficacy in supporting the user to locate specific information in documents or videos. Both tasks simulated the situation where the user has to switch back and forth between documents and videos, referencing typical scenarios of viewing on-demand lecture videos [23, 34, 35]. In a V2D task, we asked the user to scroll and search for information that was presented in a video. Conversely, in a D2V task, we asked the user to use a seek bar to find a scene in a video that corresponded to the information in a given document. Note that we did not counterbalance the order of the two exploration tasks because we did not intend to compare performance metrics between tasks with such different procedures. For these two exploration tasks, we measured task completion time, logged user operations (mouse pointer movement and scrolling), and tracked eye gaze. In addition, we collected comments from participants on their experience and areas for improvement for the user interface, as well as the System Usability Scale (SUS) score and the NASA-TLX score to evaluate the subjective usability and workload.

[15] https://react-pdf.org/

We used four existing document-based videos in the study. For the slide-based videos, We chose two presentation videos (Slide02 and Slide03 in Table 1) of previous international conferences by the author's research team. For the article-based videos, we used two videos (Article02 and Article03 in Table 1) introducing AI papers publicly available on YouTube. We selected these materials because both video data and document data (whose content is the same as in the video) were available, and the contents have similar difficulty within each type of document (slide and article). Because the durations of article-based videos were about 25 minutes each, we modified them to be about five minutes long with the permission of the video creators. Accordingly, we used the shortened article PDFs (into four pages for both Article02 and Article03) because we were concerned the difficulty of the task would become too high. Note that we did not use our sequence analyzer (described in section 3.2) and the content matching between the videos and the documents was detected manually by us before the user study. The reason for this was that we aimed to purely evaluate the concept of our interface without considering the accuracy of the sequence analyzer.

4.3 Experimental Setup

Figure 4b shows the experimental setup from a participant's viewpoint. We used a 15-inch laptop (GIGABYTE AERO 15: Core i7 9750H, 16GB, GeForce RTX 1660ti 8GB), an extended 15.6-inch mobile monitor, and a wireless mouse (Logitech M705) for the task. The laptop's main screen displayed document-based video window(s), and the sub-screen shown with the extended monitor displayed Google Form for entering answers during the task. In addition, Tobii Pro Spark was placed on the bottom of the laptop's display to measure participants' eye gaze using Tobii Pro Lab software.

4.4 Procedure

4.4.1 Overall procedure. Figure 5 shows the overall flow of the experiment. The experiment was divided into two blocks, with the fixed order of tasks using slide documents in the first half and article documents in the second half. This was to take into account the generally high difficulty of the tasks in this experiment; we wanted to make it easier for participants to become familiar with the tasks by using the slide documents (with less information) earlier, which were assumed to be easier than the article documents. We counterbalanced the order of interface presented within each block and the combination of interface and document material across the participants. Overall, the experiment consisted of four trials (2 interfaces x 2 document types) and lasted about 100 minutes per participant.

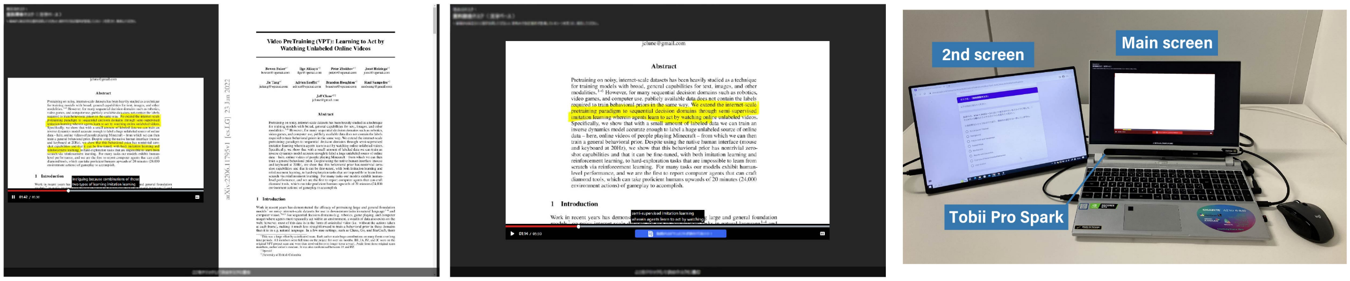
The specific procedure of the experiment was as follows. After receiving an overview of the experiment, participants began the tutorial for the first block, tasks using slide documents. In the tutorial, participants were shown how to operate each interface using a sample video material (Slide01 in Table 1), and were able to try out the interfaces until they became familiar with them. Participants then conducted actual trials with each interface (approx. 15 minutes per interface, details in the following section). After completing the trials with the two interfaces, they answered subjective questionnaires (*i.e.,* SUS and NASA-TLX) on Google Form and finally responded to semi-structured oral interviews. The interviews asked about differences in usability between the interfaces and improvements to our interface. After a short break, the second block of the experiment using article documents followed, with the overall experiment ending after this part.

4.4.2 Task procedure. The experimental tasks were performed in the following order: (1) summary task, (2) V2D task, and (3) D2V task. The details for each task are as follows.

(1) **Summary task**. Participants were asked to carefully watch a document-based video and an accompanying document using each interface, and to provide a summary of the video in free-text format on a Google Form during or after watching the video. To simulate real-time viewing, seek bar operations (pause, playback) were not available. The obtained summary in this task were not included in the evaluation because the main purpose of the task was to obtain subjective usability through actual use of each interface.
(2) **V2D task**. We asked each participant to perform two trials of V2D tasks. In a V2D task, using either the baseline interface or SwapVid, each participant was asked to scroll through the document to find a target (*i.e.,* text or an image) that fulfilled instructions derived from a particular video scene (*e.g.,* "*Regarding the slide shown in the scene at 2:45-3:19, please answer the title of the slide three pages before the slide.*", "*Regarding the section of the article shown in the video from 1:04 to 1:57, please answer whether or not the text "in the same way" is included in the section.*"). The interface automatically played the relevant part of the video upon starting the trial, while the participant was instructed to search for the corresponding section in the document. For this task (and the following D2V task), the keyword search functionality in the document viewer was disabled. Once the participant found the targeted answer, they chose it from the options presented on the Google Form, and then pressed the "complete" button in the main screen. The task completion time was measured from the time the task screen was displayed until the "complete" button was pressed.
(3) **D2V task**. Following the V2D task, the participants were asked to work on the D2V task, which consisted of two trials. The task presented participants with a target (*i.e.,* text or an image) that was part of the document and asked them to search for a video scene containing it (*e.g.,* "*Please pause the video at the scene where the text "Introduction" appears in the video and submit the task*"). The participant was informed that the recommended way to proceed with the task was to first display the relevant section in the document and then search for the corresponding video scene. After the participant found the target scene, the task was completed by pressing the "complete" button.

4.5 Results

All participants completed all trials. However, because there was an incomplete question in the first trial of V2D task with article document, we discarded the data for the corresponding 20 trials (1 trial x 20 participants). For the remaining trials, the mean correct answer rate was 81.9% (baseline: 80.0%, SwapVid: 83.8%) for V2D task and



Baseline interface

SwapVid interface

(a) Screenshots of each interface presented

(b) Apparatus

Figure 4: Experimental setup; each interface as shown in (a) was presented on the main screen in (b).

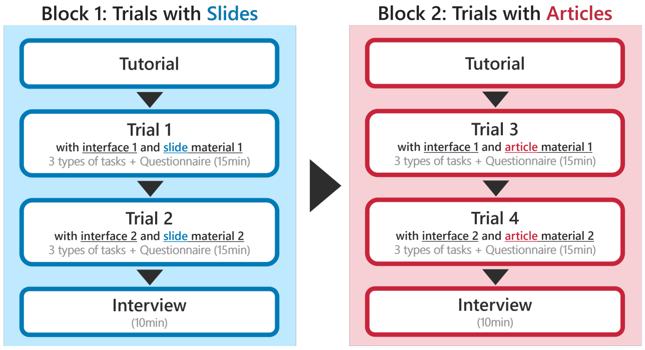

Block 1: Trials with Slides
Tutorial
Trial 1
with interface 1 and slide material 1
3 types of tasks + Questionnaire (15min)
Trial 2
with interface 2 and slide material 2
3 types of tasks + Questionnaire (15min)
Interview
(10min)
Block 2: Trials with Articles
Tutorial
Trial 3
with interface 1 and article material 1
3 types of tasks + Questionnaire (15min)
Trial 4
with interface 2 and article material 2
3 types of tasks + Questionnaire (15min)
Interview
(10min)

Figure 5: Overall flow of the experiment.

99.4% (baseline: 100%, SwapVid: 98.8%) for the D2V task. Below, we show the analysis results including incorrect trials, because most incorrect trials seemed to be due to misinterpretation of the question and we observed participants' appropriate target-seeking behavior on these trials as well. Because most of the data obtained did not show normality, we used nonparametric statistical methods in the following analyses.

4.5.1 *Task completion time.*

Figure 6a shows the results of the task completion time related to interface in each task. We performed a two-way Aligned Rank Transform [56] (ART) ANOVA on V2D task completion time, showing no main effect of interface ($F(1, 20) = 3.51, p = 0.065$). A Wilcoxon signed rank test revealed that SwapVid provides a shorter task completion time than the baseline in slide document ($W = 188.0, p = 0.0010$). For the D2V task, a two-way ART ANOVA on task completion time showed no main effect for interface ($F(1, 20) = 0.076, p = 0.784$)

4.5.2 *Cursor movement distance.*

We calculated the cursor movement distance by summing the distances between the mouse cursor positions (px[16]) obtained at regular intervals during the task. Figure 6b shows the results of the cursor movement distance related to interface in each task. A two-way ART ANOVA on cursor movement distance in V2D task revealed a main effect of interface ($F(1, 20) = 5.78, p = 0.019$). A Wilcoxon signed rank test revealed that SwapVid provided significantly less movement distance for slide ($W = 202.0, p < .001$). We then conducted a two-way ART ANOVA on cursor movement distance in D2V task, and found no main effect of interface ($F(1, 20) = 0.168, p = 0.683$). However, a Wilcoxon signed rank test showed that SwapVid provided less movement distance for slide ($W = 162.0, p = 0.033$).

4.5.3 *Scrolled amount.*

We defined a metric of scrolled amount that represents the substantial length scrolled within a document, calculated by the on-screen scrolled distance (px) divided by the height of the entire document (px) in each trial. This definition is intended to provide a fair comparison between interfaces with different displayed scale of the document (please see Figure 4a).

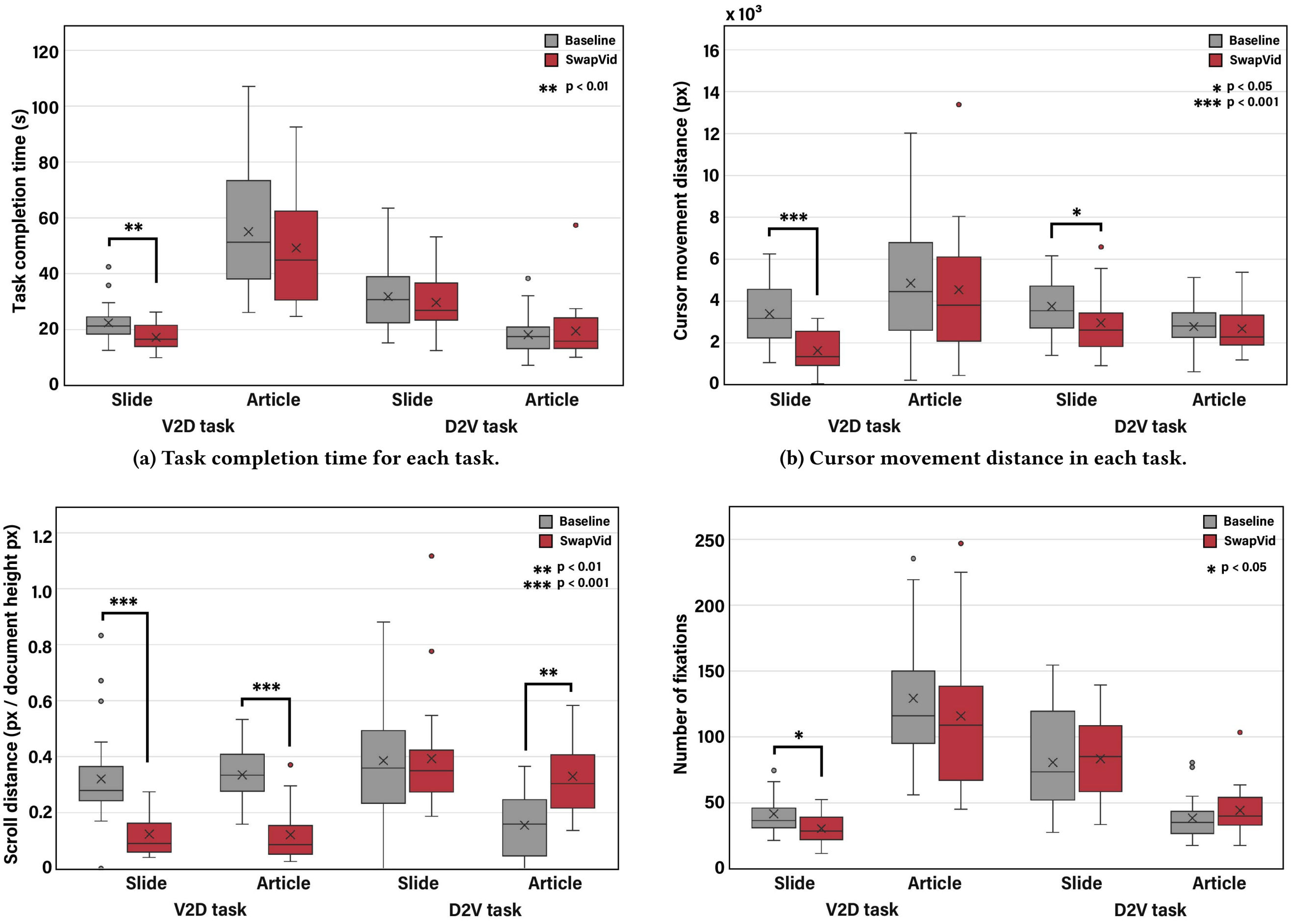
Figure 6c shows the results of the scrolled amount related to interface in each task. We performed a two-way ART ANOVA on V2D task and found a main effect of interface ($F(1, 20) = 62.53, p < .001$). A Wilcoxon signed rank test showed that SwapVid resulted in significantly less scrolling than the baseline for both slide ($W = 195.0, p < .001$) and article ($W = 203.0, p < .001$). For D2V task, a two-way ART ANOVA showed a main effect of interface ($F(1, 20) = 4.24, p = 0.043$). A Wilcoxon signed-rank test revealed that SwapVid gave significantly more scrolling in article document ($W = 24.0, p = 0.0010$), contrary to the results of V2D task.

4.5.4 *Visual fixation.*

Figure 7 shows heatmap representations of the participants' gaze (*i.e.,* fixation) distribution using each interface for the second trial of each task. Not surprisingly, the distribution for the baseline is divided into left and right regions, whereas for SwapVid, it is generally clumped into specific regions within the window.

We also examined the number of fixations that were detected by the Tobii Pro Lab software. Figure 6d shows the results of the number of fixations related to interface in each task. A two-way ART ANOVA on number of fixations in V2D task showed no main effect of interface ($F(1, 20) = 3.77, p = 0.056$). However, a Wilcoxon signed-rank test revealed that SwapVid produced significantly fewer fixations on slide document ($W = 159.5, p = 0.010$). For D2V task, a two-way ART ANOVA on number of fixations showed no main effect of interface ($F(1, 20) = 0.723, p = 0.398$). We also examined the duration of fixations and obtained similar results; a Wilcoxon signed-rank test showed that the duration with SwapVid ($7.16s \pm 2.82$) was significantly less than that with the baseline ($9.13s \pm 3.33$) for V2D task ($W = 146.0, p = 0.040$). According to the previous work on visual computing [58], the lower number and duration of

[16] The obtained "px" hereafter refers to CSS pixels.



(a) Task completion time for each task.

(b) Cursor movement distance in each task.

(c) Scrolled amount in each task.

(d) Number of fixations during each task.

Figure 6: Results of V2D and D2V tasks with box plots; the "×" marks in the graphs indicate mean values.

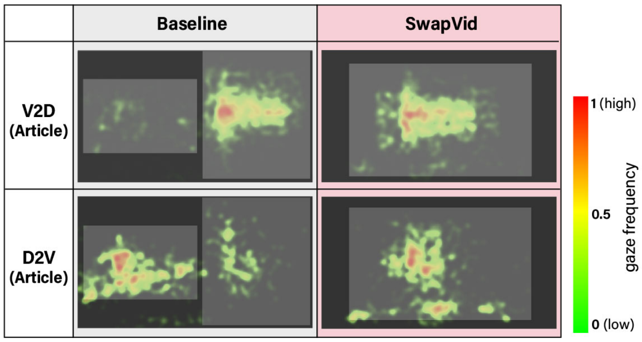


Figure 7: Participants' gaze distributions using Tobii I-VT (Fixation) filter in the second trial of each task with articles; the translucent rectangles represent the window layouts in each interface.

fixations obtained with our interface in some conditions could be interpreted as indicating a lower cognitive load on the participants.

Table 2: Obtained NASA-TLX weighted rating scores

Interface	Slide	Article
Baseline	60.10 (SD=16.3)	66.42 (SD=15.9)
SwapVid	**48.28** (SD=17.9)	**58.80** (SD=18.6)

4.5.5 Subjective workload. Table 2 shows the weighted rating of the obtained NASA-TLX scores for each condition. We performed a two-way ART ANOVA and found a main effect of interface ($F(1, 20) = 6.379, p = 0.014$). Wilcoxon signed rank tests revealed that the score was significantly better with SwapVid than the baseline for both slide ($W = 182.0, p = 0.0030$) and article ($W = 173.5, p = 0.011$).

4.5.6 Subjective usability and preferences. The mean SUS scores obtained for SwapVid were 76.25 ($SD = 14.69$) in slide document and 76.63 ($SD = 14.58$) in article document, both equivalent to "*GOOD*" in adjective rating.

Figure 8 shows the obtained user preferences for each condition. In the summary task, the baseline interface was slightly preferred by the participants over our interface in both document types. In

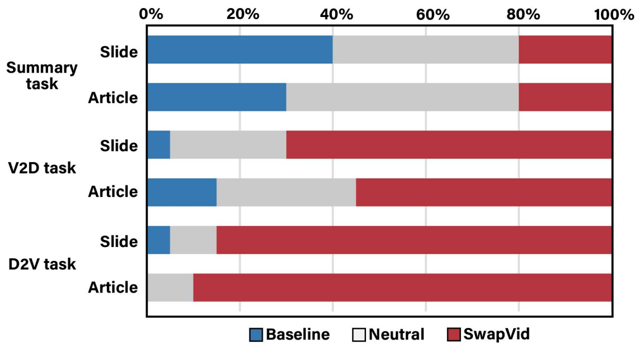


Figure 8: Subjective preferences in each task.

contrast, our interface was preferred by the majority of participants in the V2D and D2V tasks.

5 DISCUSSION

5.1 Exploration Task Performance

The results showed that the participants performed V2D tasks more effectively using SwapVid, in terms of shorter task completion time, less cursor movement distance, less scrolled page amount, less number and duration of fixations, and better user preferences for at least one of the document types. We believe these results are derived from the feature of our interface that allows users to effortlessly switch modes from video to document simply by scrolling, which substantially reduces the time and physical workload for the video-document transition. This interpretation is also supported by the participants' comments, all of which (N=20) mentioned the ease of navigation between the videos and the documents.

In contrast to V2D tasks, there were no significant improvements in performances for D2V tasks, suggesting no clear advantage of our interface over the baseline, even though many participants (N=17) noted the usefulness of the highlighting on the seek bar. This may involve two following reasons regarding our D2V task settings. First, the videos that we used were around five minutes, which seems to have made it possible for the participants to memorize the contents in the summary task (*i.e.,* the participants could find the targeted time in the video without using the highlighting feature). Many meeting and lecture videos are generally longer in duration and we do not usually know the content in advance, in which case it will be quite challenging to find relevant time without the highlighting and our interface will be more advantageous in D2V tasks. Second, due to the differences in window size and aspect ratio between the interfaces, the document viewer of the baseline interface was able to display more pages in a single window than that of our interface (please see Figure 4a). We speculate that this led to an increase in the overall workload for document exploration using our interface relative to the baseline (whereas in V2D tasks, this may have been a minor issue as the benefits of the smooth video-to-document transition feature of our interface was more substantial). Similarly, our interface provided a larger seek bar than the baseline due to the different window size, which may have resulted in longer cursor movement distances in D2V tasks. Thus, further clarification is needed to better understand how our interface could support D2V tasks.

In terms of document type, our interface generally performed better with slides than with article documents. We suspect that the characteristics of the materials of video and document used caused the difference; because the article documents' page lengths were about half that of the slides, the user could browse through the documents to find the targeted information more easily, which may have underutilized the advantage of our interface. Another possible reason is that the window aspect of our interface (*i.e.,* landscape) did not match that which would be appropriate for viewing article (*i.e.,* portrait) documents; this might have led to the higher amount of scrolling for article documents in D2V tasks using our interface (as seen in Figure 6c).

5.2 General Usability

The participants generally accepted our interface and found it useful. Although many of the participants had attended many online lectures before and were accustomed to interfaces similar to the baseline, they generally preferred our interface, and the obtained SUS score showed "*GOOD*" in either document type. This can be primarily attributed to our interface's successful blending of an existing video viewer and document viewer, requiring almost no extra effort to learn how to use.

In terms of task-specific preferences (Figure 8), our interface was dominantly preferred over the baseline for both V2D and D2V tasks, while the baseline was rather preferred in the summary task. Participants who preferred our interface more in the summary task, in the minority, mentioned the ability to view content larger than the baseline (N=9) and the ability to focus on a single view (N=5) as the reasons. However, the majority preferred the baseline that could provide document and video views individually. More specifically, many of the participants who preferred the baseline cited the following limitations in our interface when in document mode. First, the most frequently mentioned (N=14) limitation was that users could not visually check the progress of the video in the document mode. We initially thought it would be more beneficial to concentrate on either mode, but given the comments, we decided to explore an improved interface in the next section. Second, participants (N=8) also mentioned the inability to view videos and/or animations embedded in the slides when in document mode. This could be improved by embedding dynamic contents such as videos and animations into the document viewer of our interface, *e.g.,* by integrating the system of Masson et al [42]. The third limitation mentioned (N=6) was the inability to provide the document overview. Support for zooming and thumbnail views, as many document viewers have, would be an essential future work.

In terms of screen size, we used a laptop with a 15-inch display, which is one of the mainstream devices used to view document-based videos. We believe that mobile devices such as tablets and smartphones would benefit more from saving screen space for our interface, and we explore this in the following section.

6 EXPANDING SWAPVID'S PRACTICAL CAPABILITIES

Based on the user study findings, we implemented an improved SwapVid's interface with additional functionalities to expand its practical capabilities. Further polishing the tool's quality was deemed necessary as our goal was to make the tool publicly available upon publication. We obtained user feedback (N=6) on the updated interface with which we informally evaluated the usability of the new iteration of SwapVid, which included a functioning sequence analyzer.

6.1 Extension Design and Implementation

We designed and implemented an extended interface plus additional functionalities, with the three main points below. We published the source code of the entire system, including this extension, on Github[17].

6.1.1 Mode switching by text selection. We implemented a mode transition from video mode to document mode by text selection, based on the same idea as the transition by scrolling. In video mode, the cursor changes to an I-shape when it is over text, and as soon as it detects dragging of the text, the system switches to document mode and the text becomes selected. This feature can greatly support workflows such as keyword searching within the document related to the video content being viewed.

6.1.2 Picture-in-picture video view. To resolve the inability to view the video when in document mode, our interface adds a picture-in-picture (PiP) video view (as shown in Figure 1b right). Specifically, the system displays the currently playing video in a small size in the lower right corner of the screen via a transition animation at the moment of switching from video mode to document mode. This PiP video view can be moved to any position in the view by dragging, and can be re-transitioned to the video mode simply by clicking.

6.1.3 Support for mobile devices. To allow SwapVid to be used on mobile devices such as smartphones and tablets, we added support for touch operations of the interface and responsive layout. We tested it for both iOS (*i.e.,* Webkit[18]) and Android (*i.e.,* Chromium[19]) web browsers.

6.2 User Feedback

6.2.1 Overview. We obtained preliminary user feedback on the usability of the interface with the above improvements on both a laptop and a tablet device. The participants were six students in our lab (males, mean age= 22.8 ± 0.753), all of whom had experience with the previous version (*i.e.,* the interface used in the user study).

Unlike the user study in the previous section, we used an actual working prototype including the sequence analyzer, in addition to the above improvements. After a brief guidance, participants were given eight minutes each (four minutes each for a slide document and an article document) on a 15-inch laptop (the same model as in the user study) and an 11-inch tablet (iPad Pro, 2nd gen). During that, they were allowed to try the interface while viewing video materials (the six materials shown in Table 1) as they liked, and then gave comments about their experience.

6.2.2 Results. Overall, all participants found our interface useful and were willing to use it personally. Their particularly favorable points included the ease to navigate between videos and documents (N=3) and the effective use of the screen space to view both video and document on a single screen (N=3).

The updated features of our interface generally received positive comments. Regarding the PiP video view, all participants responded positively about its usefulness. However, many (N=5) were concerned about the fact that the PiP view partially occludes the document on both devices (especially on the tablet), and suggested that the PiP view could be resized or hidden as an improvement. The overall user experience on the tablet was generally as good as (N=5) or better than (N=1) that on the laptop, and most of the reasons for the positive feedback referred to the intuitiveness of the touch operation (N=3). A participant particularly appreciated that the tablet alone could complete the viewing of both videos and documents. There was also a desire for pinch-zoom support and the addition of a stylus pen annotation function on the document viewer. Regarding the text selection function, many comments (N=3) particularly appreciated the convenience and practicality of the text search. Regarding the behavior of the sequence analyzer, no participant had problems switching from video to document or complained about the visual changes during the switch, suggesting that the analyzer is practical to use.

7 LIMITATIONS AND FUTURE WORK

A major limitation of this study is that the current implementation does not work with real-time videos. The observed computation time of the sequence analyzer (Table 1) suggests the premise for real-time execution; future work will be to integrate this interface into video call applications.

Another limitation is that the sequence analyzer currently supports limited document content types and formats. In particular, since the content matching algorithm relies on OCR, it would not work with content that contains no text or is very low resolution. We will improve this by additionally using an image-based content matching approach when OCR fails, as in prior work [2]. In addition, the current implementation works with one-dimensional sequential and static documents, but we will explore ways to support documents that contain dynamic content (e.g., animations and embedded videos), such as PowerPoint slides. It is also worth exploring support for documents with links (in PDF documents or web pages) by recognizing their link structure (*e.g.,* [7]).

Future work also includes improving our interface to further make navigation between video and documents more efficient. The results of the user study suggest that the landscape window aspect of our interface may make it more difficult to explore portrait documents. In response, improvements may be needed to compensate for the mismatch in window aspect when switching from video to document, for example, by automatically zooming out, transforming the window aspect ratio, or providing a thumbnail view.

8 CONCLUSION

We proposed SwapVid, a novel interface for viewing and exploring document-based videos that allows seamless switching between a

[17] https://github.com/icd-tohoku/SwapVid_Public
[18] https://webkit.org/
[19] https://www.chromium.org/Home/

video and a document in a single view with direct manipulation. The results of the user study show that the interface helps the user reduce time and physical workload in exploring slide-based documents based on video referencing. The results also show that our interface concept is generally accepted by the users and preferred over the conventional interface they are used to. Based on these, we conclude that our interface successfully facilitates the user's concurrent exploration between a video and a document. Future work includes implementing an interface for real-time video viewing and improving the content matching algorithm to support a wide range of content types.

ACKNOWLEDGMENTS

Part of this work was performed in the Cooperative Research Project of the Research Institute of Electrical Communication, Tohoku University (R05/B14). This work is also supported by the Ministry of Education, Singapore under its Academic Research Fund Tier 2 (Project ID: T2EP20220-0016). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of the Ministry of Education, Singapore.

REFERENCES

[1] Brett Adams, Stewart Greenhill, and Svetha Venkatesh. 2012. Towards a Video Browser for the Digital Native. In *2012 IEEE International Conference on Multimedia and Expo Workshops*. 127–132. https://doi.org/10.1109/ICMEW.2012.29

[2] John Adcock, Matthew Cooper, Laurent Denoue, Hamed Pirsiavash, and Lawrence A. Rowe. 2010. TalkMiner: A Lecture Webcast Search Engine. In *Proceedings of the 18th ACM International Conference on Multimedia* (Firenze, Italy) *(MM '10)*. Association for Computing Machinery, New York, NY, USA, 241–250. https://doi.org/10.1145/1873951.1873986

[3] Pablo F Alcantarilla and T Solutions. 2011. Fast explicit diffusion for accelerated features in nonlinear scale spaces. *IEEE Trans. Patt. Anal. Mach. Intell* 34, 7 (2011), 1281–1298.

[4] Bowen Baker, Ilge Akkaya, Peter Zhokov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. 2022. Video pretraining (vpt): Learning to act by watching unlabeled online videos. *Advances in Neural Information Processing Systems* 35 (2022), 24639–24654.

[5] James Begole, Mary Beth Rosson, and Clifford A. Shaffer. 1999. Flexible Collaboration Transparency: Supporting Worker Independence in Replicated Application-Sharing Systems. *ACM Trans. Comput.-Hum. Interact.* 6, 2 (jun 1999), 95–132. https://doi.org/10.1145/319091.319096

[6] Christoph Brachmann and Rainer Malaka. 2009. Keyframe-Less Integration of Semantic Information in a Video Player Interface. In *Proceedings of the 7th European Conference on Interactive TV and Video* (Leuven, Belgium) *(EuroITV '09)*. Association for Computing Machinery, New York, NY, USA, 137–140. https://doi.org/10.1145/1542084.1542109

[7] Scott A. Carter and Laurent Denoue. 2009. SeeReader: An (Almost) Eyes-Free Mobile Rich Document Viewer. *CoRR* abs/0909.2185 (2009). arXiv:0909.2185 http://arxiv.org/abs/0909.2185

[8] Ling Chen, Gen-Cai Chen, Cheng-Zhe Xu, Jack March, and Steve Benford. 2007. EmoPlayer: A media player for video clips with affective annotations. *Interacting with Computers* 20, 1 (11 2007), 17–28. https://doi.org/10.1016/j.intcom.2007.06.003 arXiv:https://academic.oup.com/iwc/article-pdf/20/1/17/2514859/iwc20-0017.pdf

[9] Christopher Clarke, Doga Cavdir, Patrick Chiu, Laurent Denoue, and Don Kimber. 2020. Reactive Video: Adaptive Video Playback Based on User Motion for Supporting Physical Activity. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 196–208. https://doi.org/10.1145/3379337.3415591

[10] Laurent Denoue, Scott Carter, and Matthew Cooper. 2013. Content-Based Copy and Paste from Video Documents. In *Proceedings of the 2013 ACM Symposium on Document Engineering* (Florence, Italy) *(DocEng '13)*. Association for Computing Machinery, New York, NY, USA, 215–218. https://doi.org/10.1145/2494266.2494313

[11] Laurent Denoue, Scott Carter, and Matthew Cooper. 2014. Video Text Retouch: Retouching Text in Videos with Direct Manipulation. In *Adjunct Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 37–38. https://doi.org/10.1145/2658779.2659102

[12] Laurent Denoue, Scott Carter, Matthew Cooper, and John Adcock. 2013. Real-Time Direct Manipulation of Screen-Based Videos. In *Proceedings of the Companion Publication of the 2013 International Conference on Intelligent User Interfaces Companion* (Santa Monica, California, USA) *(IUI '13 Companion)*. Association for Computing Machinery, New York, NY, USA, 43–44. https://doi.org/10.1145/2451176.2451190

[13] Pierre Dragicevic, Gonzalo Ramos, Jacobo Bibliowitcz, Derek Nowrouzezahrai, Ravin Balakrishnan, and Karan Singh. 2008. Video Browsing by Direct Manipulation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08)*. Association for Computing Machinery, New York, NY, USA, 237–246. https://doi.org/10.1145/1357054.1357096

[14] Quanfu Fan, Kobus Barnard, Arnon Amir, and Alon Efrat. 2011. Robust Spatiotemporal Matching of Electronic Slides to Presentation Videos. *IEEE Transactions on Image Processing* 20, 8 (2011), 2315–2328. https://doi.org/10.1109/TIP.2011.2109727

[15] Quanfu Fan, Kobus Barnard, Arnon Amir, Alon Efrat, and Ming Lin. 2006. Matching Slides to Presentation Videos Using SIFT and Scene Background Matching. In *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval* (Santa Barbara, California, USA) *(MIR '06)*. Association for Computing Machinery, New York, NY, USA, 239–248. https://doi.org/10.1145/1178677.1178710

[16] Kazuyuki Fujita, Aoi Suzuki, Kazuki Takashima, Kaori Ikematsu, and Yoshifumi Kitamura. 2021. TiltChair: Manipulative Posture Guidance by Actively Inclining the Seat of an Office Chair. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (, Yokohama, Japan,) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 228, 14 pages. https://doi.org/10.1145/3411764.3445151

[17] Andreas Girgensohn, Frank Shipman, and Lynn Wilcox. 2011. Adaptive Clustering and Interactive Visualizations to Support the Selection of Video Clips. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval* (Trento, Italy) *(ICMR '11)*. Association for Computing Machinery, New York, NY, USA, Article 34, 8 pages. https://doi.org/10.1145/1991996.1992030

[18] S. Greenberg. 1990. Sharing Views and Interactions with Single-User Applications. In *Proceedings of the ACM SIGOIS and IEEE CS TC-OA Conference on Office Information Systems* (Cambridge, Massachusetts, USA) *(COCS '90)*. Association for Computing Machinery, New York, NY, USA, 227–237. https://doi.org/10.1145/91474.91546

[19] S. Greenberg. 1990. Sharing Views and Interactions with Single-User Applications. *SIGOIS Bull.* 11, 2–3 (mar 1990), 227–237. https://doi.org/10.1145/91478.91546

[20] Saul Greenberg. 1996. A Fisheye Text Editor for Relaxed-WYSIWIS Groupware. In *Conference Companion on Human Factors in Computing Systems* (Vancouver, British Columbia, Canada) *(CHI '96)*. Association for Computing Machinery, New York, NY, USA, 212–213. https://doi.org/10.1145/257089.257285

[21] Jens Emil Grønbæk, Banu Saatçi, Carla F. Griggio, and Clemens Nylandsted Klokmose. 2021. MirrorBlender: Supporting Hybrid Meetings with a Malleable Video-Conferencing System. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 451, 13 pages. https://doi.org/10.1145/3411764.3445698

[22] Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2010. Chronicle: Capture, Exploration, and Playback of Document Workflow Histories. In *Proceedings of the 23nd Annual ACM Symposium on User Interface Software and Technology* (New York, New York, USA) *(UIST '10)*. Association for Computing Machinery, New York, NY, USA, 143–152. https://doi.org/10.1145/1866029.1866054

[23] Philip J. Guo, Juho Kim, and Rob Rubin. 2014. How Video Production Affects Student Engagement: An Empirical Study of MOOC Videos. In *Proceedings of the First ACM Conference on Learning @ Scale Conference* (Atlanta, Georgia, USA) *(L@S '14)*. Association for Computing Machinery, New York, NY, USA, 41–50. https://doi.org/10.1145/2556325.2566239

[24] Yukai Hoshikawa, Kazuyuki Fujita, Kazuki Takashima, Morten Fjeld, and Yoshifumi Kitamura. 2022. RedirectedDoors: Redirection While Opening Doors in Virtual Reality. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 464–473. https://doi.org/10.1109/VR51125.2022.00066

[25] Shengyi Huang, Santiago Ontañón, Chris Bamford, and Lukasz Grela. 2021. Gym-µRTS: Toward Affordable Full Game Real-time Strategy Games Research with Deep Reinforcement Learning. In *2021 IEEE Conference on Games (CoG)*. 1–8. https://doi.org/10.48550/arXiv.2105.13807

[26] Marco A. Hudelist, Klaus Schoeffmann, and Laszlo Boeszoermenyi. 2013. Mobile Video Browsing with the ThumbBrowser. In *Proceedings of the 21st ACM International Conference on Multimedia* (Barcelona, Spain) *(MM '13)*. Association for Computing Machinery, New York, NY, USA, 405–406. https://doi.org/10.1145/2502081.2502242

[27] W. Hurst and P. Jarvers. 2005. Interactive, dynamic video browsing with the zoomslider interface. In *2005 IEEE International Conference on Multimedia and Expo*. 4 pp.–. https://doi.org/10.1109/ICME.2005.1521484

[28] Wolfgang Hürst and Konrad Meier. 2008. Interfaces for Timeline-Based Mobile Video Browsing. In *Proceedings of the 16th ACM International Conference on*

Multimedia (Vancouver, British Columbia, Canada) *(MM '08)*. Association for Computing Machinery, New York, NY, USA, 469–478. https://doi.org/10.1145/1459359.1459422

[29] Wolfgang Hürst, Rob van de Werken, and Miklas Hoet. 2015. A Storyboard-Based Interface for Mobile Video Browsing. In *MultiMedia Modeling*, Xiangjian He, Suhuai Luo, Dacheng Tao, Changsheng Xu, Jie Yang, and Muhammad Abul Hasan (Eds.). Springer International Publishing, Cham, 261–265.

[30] Dan Jackson, James Nicholson, Gerrit Stoeckigt, Rebecca Wrobel, Anja Thieme, and Patrick Olivier. 2013. Panopticon: A Parallel Video Overview System. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) *(UIST '13)*. Association for Computing Machinery, New York, NY, USA, 123–130. https://doi.org/10.1145/2501988.2502038

[31] Thorsten Karrer, Malte Weiss, Eric Lee, and Jan Borchers. 2008. DRAGON: A Direct Manipulation Interface for Frame-Accurate in-Scene Video Navigation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08)*. Association for Computing Machinery, New York, NY, USA, 247–250. https://doi.org/10.1145/1357054.1357097

[32] Juho Kim. 2013. Toolscape: Enhancing the Learning Experience of How-to Videos. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems* (Paris, France) *(CHI EA '13)*. Association for Computing Machinery, New York, NY, USA, 2707–2712. https://doi.org/10.1145/2468356.2479497

[33] Jeongyeon Kim, Yubin Choi, Minsuk Kahng, and Juho Kim. 2022. FitVid: Responsive and Flexible Video Content Adaptation. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 501, 16 pages. https://doi.org/10.1145/3491102.3501948

[34] Juho Kim, Philip J. Guo, Carrie J. Cai, Shang-Wen (Daniel) Li, Krzysztof Z. Gajos, and Robert C. Miller. 2014. Data-Driven Interaction Techniques for Improving Navigation of Educational Videos. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 563–572. https://doi.org/10.1145/2642918.2647389

[35] Juho Kim, Philip J. Guo, Daniel T. Seaton, Piotr Mitros, Krzysztof Z. Gajos, and Robert C. Miller. 2014. Understanding In-Video Dropouts and Interaction Peaks Inonline Lecture Videos. In *Proceedings of the First ACM Conference on Learning @ Scale Conference* (Atlanta, Georgia, USA) *(L@S '14)*. Association for Computing Machinery, New York, NY, USA, 31–40. https://doi.org/10.1145/2556325.2566237

[36] Tae Soo Kim, Matt Latzke, Jonathan Bragg, Amy X. Zhang, and Joseph Chee Chang. 2023. Papeos: Augmenting Research Papers with Talk Videos. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (<conf-loc>, <city>San Francisco</city>, <state>CA</state>, <country>USA</country>, </conf-loc>) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 15, 19 pages. https://doi.org/10.1145/3586183.3606770

[37] Don Kimber, Tony Dunnigan, Andreas Girgensohn, Frank Shipman, Thea Turner, and Tao Yang. 2007. Trailblazing: Video Playback Control by Direct Object Manipulation. In *2007 IEEE International Conference on Multimedia and Expo*. 1015–1018. https://doi.org/10.1109/ICME.2007.4284825

[38] Francis C. Li, Anoop Gupta, Elizabeth Sanocki, Li-wei He, and Yong Rui. 2000. Browsing Digital Video. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (The Hague, The Netherlands) *(CHI '00)*. Association for Computing Machinery, New York, NY, USA, 169–176. https://doi.org/10.1145/332040.332425

[39] Nan Li, Łukasz Kidziński, and Pierre Dillenbourg. 2015. Augmenting Collaborative MOOC Video Viewing with Synchronized Textbook. In *Human-Computer Interaction – INTERACT 2015*, Julio Abascal, Simone Barbosa, Mirko Fetter, Tom Gross, Philippe Palanque, and Marco Winckler (Eds.). Springer International Publishing, Cham, 81–88.

[40] Yujia Li, David Choi, Junyoung Chung, Nate Kushman, Julian Schrittwieser, Rémi Leblond, Tom Eccles, James Keeling, Felix Gimeno, Agustin Dal Lago, et al. 2022. Competition-level code generation with alphacode. *Science* 378, 6624 (2022), 1092–1097.

[41] Ching Liu, Juho Kim, and Hao-Chuan Wang. 2018. ConceptScape: Collaborative Concept Mapping for Video Learning. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3173574.3173961

[42] Damien Masson, Sylvain Malacria, Edward Lank, and Géry Casiez. 2020. Chameleon: Bringing Interactivity to Static Digital Documents. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376559

[43] Matthew K Miller, Frederik Brudy, Tovi Grossman, George W. Fitzmaurice, and Fraser Anderson. 2023. Peek-At-You: An Awareness, Navigation, and View Sharing System for Remote Collaborative Content Creation. In *Graphics Interface 2023 - second deadline*. https://openreview.net/forum?id=U8p66V2PeEa

[44] Cuong Nguyen and Feng Liu. 2015. Making Software Tutorial Video Responsive. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 1565–1568. https://doi.org/10.1145/2702123.2702209

[45] Cuong Nguyen, Yuzhen Niu, and Feng Liu. 2013. Direct Manipulation Video Navigation in 3D. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 1169–1172. https://doi.org/10.1145/2470654.2466150

[46] Cuong Nguyen, Yuzhen Niu, and Feng Liu. 2014. Direct Manipulation Video Navigation on Touch Screens. In *Proceedings of the 16th International Conference on Human-Computer Interaction with Mobile Devices & Services* (Toronto, ON, Canada) *(MobileHCI '14)*. Association for Computing Machinery, New York, NY, USA, 273–282. https://doi.org/10.1145/2628363.2628365

[47] Kumpei Ogawa, Kazuyuki Fujita, Kazuki Takashima, and Yoshifumi Kitamura. 2022. PseudoJumpOn: Jumping onto Steps in Virtual Reality. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 635–643. https://doi.org/10.1109/VR51125.2022.00084

[48] Amy Pavel, Dan B. Goldman, Björn Hartmann, and Maneesh Agrawala. 2015. SceneSkim: Searching and Browsing Movies Using Synchronized Captions, Scripts and Plot Summaries. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (Charlotte, NC, USA) *(UIST '15)*. Association for Computing Machinery, New York, NY, USA, 181–190. https://doi.org/10.1145/2807442.2807502

[49] Amy Pavel, Colorado Reed, Björn Hartmann, and Maneesh Agrawala. 2014. Video Digests: A Browsable, Skimmable Format for Informational Lecture Videos. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 573–582. https://doi.org/10.1145/2642918.2647400

[50] Suporn Pongnumkul, Mira Dontcheva, Wilmot Li, Jue Wang, Lubomir Bourdev, Shai Avidan, and Michael F. Cohen. 2011. Pause-and-Play: Automatically Linking Screencast Video Tutorials with Applications. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. Association for Computing Machinery, New York, NY, USA, 135–144. https://doi.org/10.1145/2047196.2047213

[51] Suporn Pongnumkul, Jue Wang, and Michael Cohen. 2008. Creating Map-Based Storyboards for Browsing Tour Videos. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology* (Monterey, CA, USA) *(UIST '08)*. Association for Computing Machinery, New York, NY, USA, 13–22. https://doi.org/10.1145/1449715.1449720

[52] Klaus Schoeffmann and Lukas Burgstaller. 2015. Scrubbing Wheel: An Interaction Concept to Improve Video Content Navigation on Devices with Touchscreens. In *2015 IEEE International Symposium on Multimedia (ISM)*. 351–356. https://doi.org/10.1109/ISM.2015.20

[53] M. Stefik, D. G. Bobrow, G. Foster, S. Lanning, and D. Tatar. 1987. WYSIWIS Revised: Early Experiences with Multiuser Interfaces. *ACM Trans. Inf. Syst.* 5, 2 (apr 1987), 147–167. https://doi.org/10.1145/27636.28056

[54] Marie-luce Viaud, Olivier Buisson, Agnes Saulnier, and Clement Guenais. 2010. Video Exploration: From Multimedia Content Analysis to Interactive Visualization. In *Proceedings of the 18th ACM International Conference on Multimedia* (Firenze, Italy) *(MM '10)*. Association for Computing Machinery, New York, NY, USA, 1311–1314. https://doi.org/10.1145/1873951.1874209

[55] Feng Wang, Chong-Wah Ngo, and Ting-Chuen Pong. 2003. Synchronization of Lecture Videos and Electronic Slides by Video Text Analysis. In *Proceedings of the Eleventh ACM International Conference on Multimedia* (Berkeley, CA, USA) *(MULTIMEDIA '03)*. Association for Computing Machinery, New York, NY, USA, 315–318. https://doi.org/10.1145/957013.957080

[56] Jacob O. Wobbrock, Leah Findlater, Darren Gergle, and James J. Higgins. 2011. The Aligned Rank Transform for Nonparametric Factorial Analyses Using Only Anova Procedures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 143–146. https://doi.org/10.1145/1978942.1978963

[57] Chengpei Xu, Ruomei Wang, Shujin Lin, Xiaonan Luo, Baoquan Zhao, Lijie Shao, and Mengqiu Hu. 2019. Lecture2Note: Automatic Generation of Lecture Notes from Slide-Based Educational Videos. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*. 898–903. https://doi.org/10.1109/ICME.2019.00159

[58] Johannes Zagermann, Ulrike Pfeil, and Harald Reiterer. 2016. Measuring Cognitive Load Using Eye Tracking Technology in Visual Computing. In *Proceedings of the Sixth Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization* (Baltimore, MD, USA) *(BELIV '16)*. Association for Computing Machinery, New York, NY, USA, 78–85. https://doi.org/10.1145/2993901.2993908

[59] Xiangrong Zhang, Chen Li, Shang-Wen Li, and Victor Zue. 2016. Automated Segmentation of MOOC Lectures towards Customized Learning. In *2016 IEEE 16th International Conference on Advanced Learning Technologies (ICALT)*. 20–22. https://doi.org/10.1109/ICALT.2016.25

[60] Baoquan Zhao, Songhua Xu, Shujin Lin, Ruomei Wang, and Xiaonan Luo. 2019. A New Visual Interface for Searching and Navigating Slide-Based Lecture Videos. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*. 928–933. https://doi.org/10.1109/ICME.2019.00164