DynamicSlide: Exploring the Design Space of Reference-based Interaction Techniques for Slide-based Lecture Videos

Hyeungshik Jung
School of Computing, KAIST
Daejeon, Republic of Korea
hyeungshik.jung@kaist.ac.kr

Hijung Valentina Shin
Adobe Research
Cambridge, Massachusetts
vshin@adobe.com

Juho Kim
School of Computing, KAIST
Daejeon, Republic of Korea
juhokim@kaist.ac.kr

ABSTRACT

Slide-based video is a popular format of online lecture videos. Lecture slides and narrations are complementary: while slides visually convey the main points of the lecture, narrations add detailed explanations to each item in the slide. We define a pair of slide item and the relevant sentence in the narration as a *reference*. In order to explore the design space of reference-based interaction techniques, we present DynamicSlide, a video processing system that automatically extracts references from slide-based lecture videos and a video player leveraging these references. Through participatory design workshops, we elicited useful interaction techniques specifically for slide-based lecture videos. Among these ideas, we selected and implemented three reference-based techniques: emphasizing the current item in the slide that is being explained, enabling item-based navigation, and enabling item-based note-taking. We also built a pipeline for automatically identifying references, and it shows 79% accuracy for finding 141 references in five videos from four different authors. Results from a user study suggest that DynamicSlide's features improve the learner's video browsing and navigation experience.

KEYWORDS

Educational videos; Visual navigation; Video learning; Multimedia learning; Video content analysis.

ACM Reference Format:
Hyeungshik Jung, Hijung Valentina Shin, and Juho Kim. 2018. DynamicSlide: Exploring the Design Space of Reference-based Interaction Techniques for Slide-based Lecture Videos. In *2018 Workshop on Multimedia for Accessible Human Computer Interface (MAHCI'18), October 22, 2018, Seoul, Republic of Korea.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3264856.3264861

1 INTRODUCTION

Lecture videos play a crucial role in today's online learning. Major Massive Open Online Course (MOOC) providers, commercial educational platforms, and individual instructors use videos as their primary course material. There are diverse styles of lecture videos, such as screencasts, Khan Academy-style, classroom recordings, slide-based lectures, and blackboard-style [7]. Among these, slide-based lectures are widely used for their familiarity, abundant pre-existing materials [31], and ease of sharing with students.

Slides are designed to visually convey the lecture material, and slide contents are closely related to the instructor's verbal narration. Slides and narrations complement each other: while slides provide a visual, structured summary of the lecture, narrations usually provide more detailed explanations on each topic represented in the slide. For learners, understanding the relationship between the slide contents and the accompanying narration is key to comprehending the lecture. While both instructors and learners actively seek this link between slides and narrations when authoring and watching lecture videos, existing video players do not expose this information in a way that supports the cognitively demanding task that both encounter.

In this paper, we explore interaction techniques that expose and leverage the relationship between slide contents and narrations to improve the learner's video watching experience. We refer to this relationship as *references*. Specifically, we define a *reference* as a link between a text item in the slide and the corresponding sentence from the narration. References provide a discretized representation of the video, which can be used to enable effective interactions, such as non-linear navigation, content search, or editing. Through a design workshop with learners, we explore the wide design space of reference-based video interaction techniques, and select a set of features to implement.

To assess the feasibility of these ideas, we built DynamicSlide, a video processing system and a video player for slide-based lecture videos. Given an input lecture video, DynamicSlide extracts the slides used in the lecture, detects the title and text segments within each slide, and computes the correspondence between each text segment and parts of the narration. With these references, the DynamicSlide player automatically emphasizes the currently explained text item in the slide, supports item-based navigation, and enables efficient re-watching of video by allowing users to bookmark items with links to the relevant parts of the video.

We evaluated the accuracy of our algorithmic pipeline by comparing automatically generated results against manually constructed ground truth results. For each of the three stages in the pipeline, slide boundary detection (Stage 1) shows 75% accuracy, text segmentation within slides (Stage 2) shows 67% accuracy, and text-to-script alignment (Stage 3) correctly finds 79% of the references using ground truth result of previous steps as the input data. In a preliminary user study, we observed how participants use DynamicSlide's features for interacting with lecture videos. We also investigated participants' preference for different methods of highlighting the

MAHCI'18, October 22, 2018, Seoul, Republic of Korea

ACM ISBN 978-1-4503-5980-1/18/10…$15.00
https://doi.org/10.1145/3264856.3264861

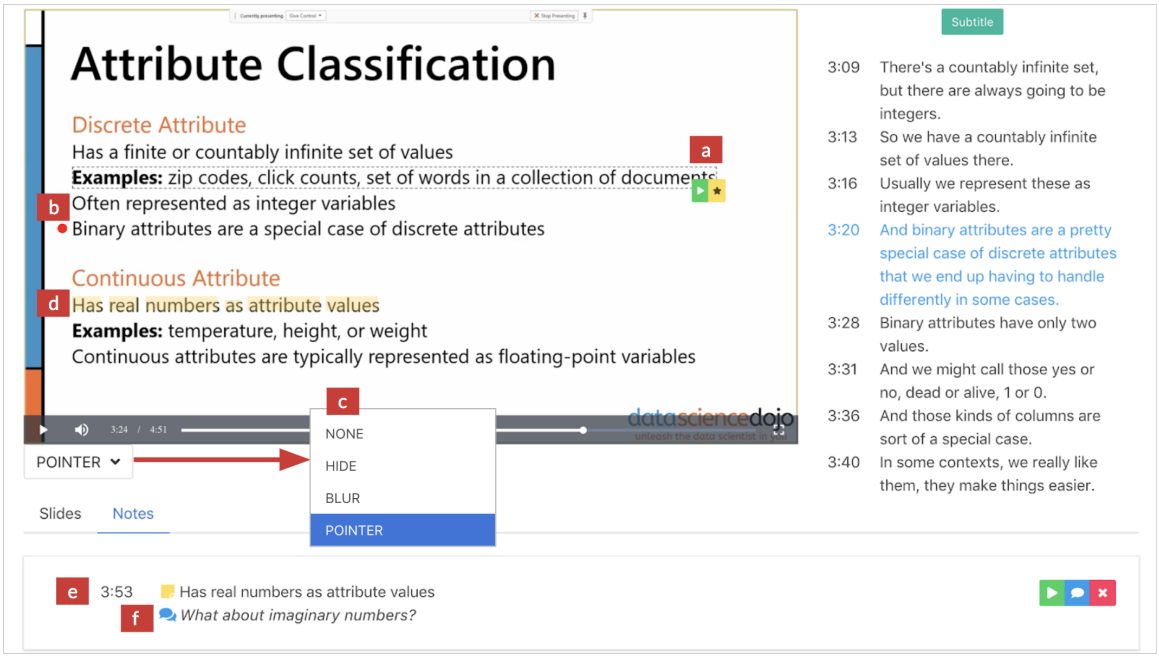

Figure 1. DynamicSlide recovers and uses the link information between a slide item and the corresponding verbal explanation in a lecture video to enable a set of reference-based interaction techniques. Main features of DynamicSlide's video player: (a) Each text item in the slide has a play button and a note-taking button that appear on hovering. (b) The current item explained by the instructor is emphasized by an indicator symbol (red dot). (c) Users can choose between three options for emphasizing the current item, which is immediately applied to the video. (d) Users can directly highlight text on a video frame. (e) Noted text items are copied and collected in a separate notes section with links to the relevant parts of the video. (f) Users can add comments to noted text items.

currently explained item of a slide. Participants found the highlighting helpful for connecting the slides and the verbal explanation as well as searching content in the video, and expected the bookmark feature to be useful in real learning scenarios.

The main contributions of this paper are as follows:

- A set of 37 interaction technique ideas for learning with slide-based lecture videos, extracted from four workshop sessions.
- DynamicSlide, a prototype video processing system and player that instantiates reference-based video interaction techniques
- Results from a user study on the effectiveness of DynamicSlide for watching and navigating slide-based lecture videos.

2 RELATED WORK

DynamicSlide draws on findings and techniques from previous research on browsing support for informational videos, object-based video interaction techniques, and methods for aligning scripts to presentation slides.

2.1 Players for Navigating Informational Videos

A thread of research has used visual and textual (transcript) information to enhance the browsing experience of information-rich videos. Video Digests [29] segmented and organized informational videos into chapters and sections. LectureScape [19] leveraged the clickstream navigation history of learners of a video to support non-linear navigation. MMToC [4] automatically created table of contents from videos, and ViZig [37] suggested video browsing based on anchor points such as charts and tables, which act as cues for navigating the video. Yang et al. [40] and Zhao et al. [41] leveraged the structure of slides as well as the script to browse videos at multiple levels of detail. Our work also exploits the relationship between slide content and scripts, but we explore a broader design space of interaction techniques in addition to navigation.

2.2 Leveraging In-video Objects for Video Interaction

Researchers have suggested using objects in videos as UI elements that users can interact with. NoteVideo [24] used hand-drawn visual elements in blackboard-style lecture videos as a UI control for video navigation. Waken [3] and Nguyen et al. [25] detected click events in software tutorial videos, and visualized the event points for supporting event-based navigation. Denoue et al. [10] made text in videos selectable and scrollable with OCR technology, and Nguyen et al. [26] analyzed the layout of lecture videos and enabled each item of the slide to be highlighted by the users' gaze. Codetube [30], Ace [38], and Codemotion [18] extracted example source codes in videos to present relevant StackOverflow questions and snippets from other programming educational videos. Extending this line of research, DynamicSlide supports video navigation and

manipulation by allowing users to interact with in-video objects that are linked to relevant parts of the script.

2.3 Slide-Script Alignment

Researchers have worked on tracking the progress of presentations by analyzing text in the slide along with the presentation narration. The core idea is to measure the similarity between text from the slide and from the narration. Yomamoto et al. [39] segmented lecture videos into topic-based chunks using topics obtained from a textbook and associated the transcript of a lecture with each topic. Lu et al. [22] aligned contents of the slide with the script with various methods, including word-based matching, structured SVM, and score integration. Asadi et al. [2] tracked the coverage of slide notes with the actual presentation in real time, by spotting keywords scored using TF-IDF and word embeddings [23]. Most related to our technique is the work by Tsujimura et al. [35], which aligns a lecture audio with a presentation file to identify the current explanation spot. DynamicSlide applies slide tracking technology to find links between slides and scripts on top of existing lecture videos, without requiring source slides. Unlike previous works, our system doesn't require an original presentation file, which means that it can be applied to a wide range of slide-based lecture videos available online.

3 DESIGN GOALS

To explore the design space of interaction techniques for slide-based lecture videos, we hosted a series of design workshops. The goal of these workshops was to identify learners' demands for video interaction and to elicit new design ideas. Participants were first introduced to state-of-the-art interaction techniques for lecture videos. Then, after watching two slide-based lecture videos, they brainstormed novel interaction techniques in individual and group ideation sessions.

3.1 Design Workshop

We recruited 12 undergraduate students (3 females) at a university for 4 sessions, each with 3 participants. All participants had experience watching lecture videos online for the university's flipped classroom courses or on YouTube. Each session took about 80 minutes. In order to familiarize participants with state-of-the-art video interaction techniques, we first introduced five previously proposed techniques from research literature, designed specifically to improve the learning experience of lecture videos: (1) gaze-based note-taking [26], (2) leveraging click-stream navigation history [19], (3) visually marking confusing points of a lecture on the video [14], (4) automatically generating static lecture notes from blackboard-style lecture videos [32], and (5) hierarchically summarizing videos into chapters and sections [29]. Participants watched a brief video or read a summary description of each technique from the project websites. Afterwards, participants watched two slide-based lecture videos, one about cryptocurrency [5] and another about political science [33], for 10 minutes each. These videos featured static slides with minimal animation, and our expectation was that participants would be able to brainstorm ideas to improve the watching experience of these baseline videos. Finally, participants were asked to brainstorm interaction ideas individually for 10 minutes. They were encouraged to think of novel features that would help them in learning, regardless of technical feasibility. Participants shared and discussed their ideas with other participants for 10 minutes, and then did another round of ideation on their own by building upon ideas from the group.

3.2 Ideas from Participants

Participants submitted a total of 53 ideas (average 4.4 ideas per participant, min=2, max=7). Our research team merged similar ideas to 37 ideas, then grouped the ideas into three categories, namely *Video Improvement*, *Video Interactions*, and *Video Augmentation*. Out of 37 ideas, we again chose 18 ideas that has technical challenges (e.g. We didn't chose ideas like showing learning goal)

3.2.1 Video Improvement. Features in this category are about making changes to how the audio or visual content of the video is played in order to improve the lecture quality.

Participatory video editing: Allow learners to fix typos or adjust awkward animations in the video.

Improve legibility of handwritten characters: Replace handwritten characters with typeface fonts (Similar to Vidwiki [8]).

Highlight the current item: Highlight the part of a material the instructor is explaining, without manual labor of instructors.

Segment a lecture video into single slide clips: Segment a long lecture video with multiple slides into shorter clips, with each clip covering a single slide and its narration. (Similar to [4])

Improve instructor's intonation: Adjust the intonation of the instructor's narration without changing its content to make it more engaging.

Automatic playback speed adjustment: Slow down the video's playback speed when the instructor is explaining a complex concept (Similar to Speeda [17]).

Improve audio quality: Improve the instructor's voice quality, for example, by reducing background noise.

3.2.2 Video Interactions. Features in this category do not directly change the audio or visual content of the video per se, but instead allow users to interact with the video content with more control.

In-video highlighting: Allow users to highlight sections of text in the slide similar to highlighting texts in documents (Similar to supporting direct manipulation of screen-based videos [10]).

Text-based navigation: Navigate a video by clicking a text segment in a slide, which would point the video to the time where that text segment is explained (Similar to NoteVideo [24]).

Zoom figures and images: Provide options to zoom in on figures and images within the video.

Provide hyperlinks to external resources: Hyperlink text and images in the slide to relevant external resources, so users can click on them for additional information.

Toggle instructor's face: Add an option to toggle the visibility of the instructor's face.

3.2.3 Video Augmentation. Features in this category are about augmenting the original video by adding additional information related to the lecture content.

Keep learners focused on the video: Alert learners when they are distracted (similar to Xiao et al. [36]).

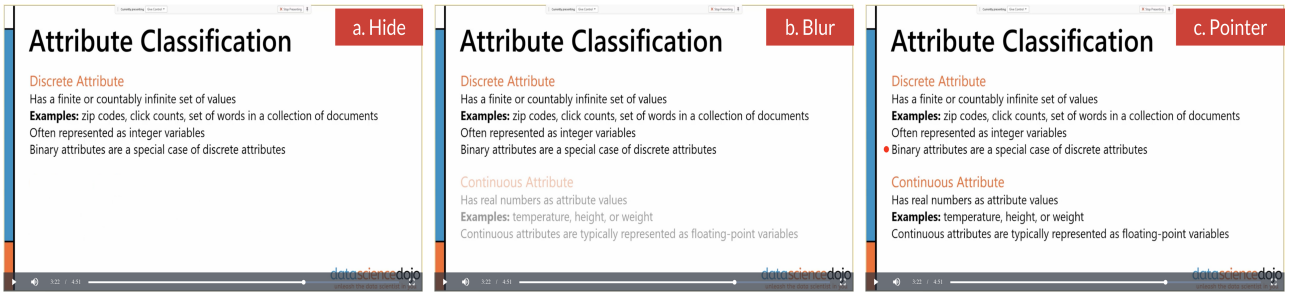

Figure 2: DynamicSlide implements three options for indicating the currently explained item in the slide. (a) *Hide* masks all future items, (b) *Blur* dims all future items, and (c) *Pointer* adds an indicator symbol next to the current item.

Show other users' navigation patterns: Show where other users paused or re-watched frequently in a video (Similar to LectureScape [19]).

Show a concept map of the lecture content: Show a concept map of a lecture video by using terms included in the slides (similar to ConceptScape [21])

Co-reference resolution: Show which entity in the video that pronouns like 'the equation,' 'the figure' point to.

Add an in-video panel for searching external resources: Search external resources related to the concept covered in the video at the moment without switching tabs.

Insert adequate memes: Show appropriate memes in the video, for example, when the instructor tells a joke.

We assessed all of the ideas contributed by the participants and narrowed down to a small set of features by the following criteria: (1) While none of the ideas are completely novel, focus on ideas that have been relatively less explored by previous research and commercial systems; (2) Focus on ideas that could benefit from the connection between visual and audio contents (i.e., references); and (3) Focus on a set of ideas that could be powered by a shared data or algorithmic pipeline.

This selection process led us to the following three design features:

(1) Add synchronous emphasis effects on the slide to indicate which item in the slide is currently being referenced by the narration (Video improvement).
(2) Enable item-based navigation by allowing users to click on parts of the slide to jump to the relevant part of the video (Video interaction).
(3) Support item-based note-taking by allowing users to copy parts of the slide to separate notes. Copied items preserve links to relevant parts of the video and can be used for navigation. (Video interaction).

4 REFERENCE-BASED INTERACTION TECHNIQUES

In this section, we introduce the DynamicSlide player, a web-based video browsing interface that incorporates the three reference-based interaction techniques discussed above. All the interaction techniques were implemented by placing DOMs over texts in the video player.

4.1 Emphasize the Current Item

The player informs the user of the current part of the slide being explained by the instructor. Each item within a slide is emphasized when the sentence referencing the item starts. The system implements three common practices of instructors for emphasizing the current item: make the slide item visible simultaneous to the explanation (Figure 2(a)), emphasize the slide item for example by brightening its color (Figure 2(b)), or place an indicator (e.g., red dot) near the item, which is analogous to using a laser pointer (Figure 2(c)).

4.2 Item-based Navigation

With item-based navigation, the user can navigate the video by selecting a particular text-based item on the slide. Extending dynamic manipulation techniques for videos [3, 25] hovering over a text item in the slide reveals two action buttons next to it (Figure 1(a)): play and bookmark. By clicking the play button, users can navigate to the starting point of the sentence that is paired with the text item. This content-based navigation enables precise playback control at the item level, which provides a significant advantage over conventional linear navigation.

4.3 Item-based Note-taking

The user can also bookmark text items in the slide. Bookmarked items are highlighted in the video and copied to a separate Notes panel (Figure 1(e)). Copied items preserve their link to the relevant parts of the video, so users can also use them for navigation. Text in the slide is copied over, so that users can edit or copy the text as needed. Finally, users can add custom notes to any bookmarked item as well (Figure 1(f)).

5 ALGORITHMIC PIPELINE

DynamicSlide's video processing pipeline takes a slide-based video as input, and returns a set of unique shots, a set of text items in each slide, and matches between text items and the narration script as output. Its goal is to automatically generate the necessary data to power the reference-based interaction techniques. The pipeline consists of three main components: (1) shot boundary detection, (2) text segmentation within slides, and (3) text-to-script alignment. Figure 3 shows an overview of the pipeline.

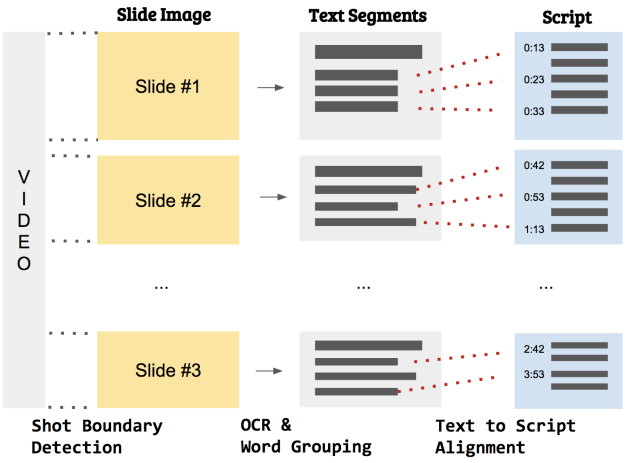

Figure 3: An overview of our automatic pipeline for finding references from slide-based lecture videos. Given an input video, Stage 1 detects unique slides with a shot boundary detection algorithm, Stage 2 finds text segments in each slide by grouping words together, and Stage 3 finds references by aligning text segments and sentences in the script.

5.1 Stage 1: Shot Boundary Detection

The purpose of this stage is to extract the set of shots used in a lecture video, and to segment the video based on the shot boundaries. We detect boundaries between unique slides, and between slide and non-slide shots (e.g., headshot of the lecturer). We use both pixel and text information from frames. We use FFmpeg [13] for frame extraction and Google Cloud Vision API [15] for text recognition.

In order to measure the difference between two consecutive frames, we use a variant of the methods suggested by Biswas et al. [4] and Zhao et al. [41]. Both methods use a criterion based on (1) image difference and (2) text difference for robustness. To measure visual change between images, we used edge-based frame difference [1] used in Zhao et al. [41] as image difference. For changes in text, we used the Levenshtein distance [20] divided by the average length of the texts extracted from the slides in two consecutive frames. Text from each slide consists of the concatenation of all the words in the slide found by the Google Cloud Vision API. [15]. We regard two frames as different if both the image difference and the text difference between two frames are higher than pre-defined thresholds (edge-based difference: 0.1, Levenshtein distance: 0.6). If a transition between shots takes more than a second multiple shot boundaries will be detected in a series. In this case, we regard only the final boundary as the true boundary.

5.2 Stage 2: Text Segmentation within Slides

The purpose of this stage is to group words in the slide into a set of semantic units, such as a phrase or a sentence. The main idea is to use the position information of words, inspired by the method used by Zhao et al. [41]. Since English text is written horizontally, we merge individual words in the slide horizontally first, then merge lines vertically for multi-line semantic units.

In horizontal merging, we merge two nearby words if their bounding boxes are horizontally aligned (i.e., they have similar top and bottom coordinates), and the horizontal distance, or the gap between the two bounding boxes, is smaller than sum of their heights. Using line heights instead of a fixed parameter is more robust to various sizes of text. This process continues until all the words on the same line are grouped together.

After grouping words into horizontal lines, we merge two lines vertically if they belong to a single semantic unit. We merge a line with the line below when the following conditions are met: (1) if the vertical distance between two lines are smaller than a threshold (set to 10px) or (2) if bounding boxes of adjacent text lines are left-aligned, center-aligned, or right-aligned on the slide. (3) Since a bullet point usually starts on a new line and indicates a separate semantic unit, if a line has a bullet point on the left, we do not merge it with the previous line. We identify bullet points by matching a pre-registered circle template after converting a slide image to grayscale with OpenCV [28].

5.2.1 Slide Title Extraction. After grouping words into semantic units, we detect and separate out the slide title or headline. Since titles mostly convey the overall theme of the slide rather than a particular point, we exclude the title from candidate text items for finding references. To find the title of a slide, we use a set of heuristics suggested by Che et al. [6].

5.3 Stage 3: Text-to-Script Alignment

The goal of this final stage is to find an alignment between the text items in a slide (identified in Stage 2) and sentences from the narration script. The main idea is to find the most textually similar sentence in the script for each text item in the slide. In this work, we use a sentence as a minimum information unit in the script. We extract sentences from subtitles using NLTK's sentence tokenizer, after restoring punctuations using an LSTM based punctuation restorator [34].

We first compute the pairwise distance between each text item (excluding the slide title) and each sentence spoken while the slide is shown. Since subtitles generated by YouTube doesn't include punctuations indicating the end of a sentence, we used a recurrent neural network based punctuation restoring model [34]. Then we obtain sentences using NLTK's sentence tokenizer. The start and end times of each sentence are defined by the start and end time of the words in the sentence, which are retrieved using the Gentle forced aligner [27].

We represent each text item and script sentence as a bag-of-words vector (weighted by the TF-IDF score of each word), after removing stop-words and tokenizing the text. We calculate the cosine similarity between these vectors as the similarity between a text item and a script sentence. Finally, for each text item in a slide, we match the most similar sentence from the script as its pair sentence. The text item - sentence pair forms a reference.

6 PIPELINE EVALUATION

We evaluate the effectiveness of each stage of our automatic pipeline. We selected nine videos that span different subjects (statistics,

journalism, data science, biology, computer science), and ran our pipeline to generate references from the videos. Table 1 describes the videos used for the evaluation. Video 1 and 5 are from MOOCs, while other videos are accessible on YouTube.

ID	Title	Length
1	Risk, variation, uncertainty [1]	11:09
2	Biology and the tree of life [2]	18:51
3	Data Attributes (Part1) [3]	4:51
4	Data Attributes (Part2) [4]	3:36
5	Solution based Journalism [5]	9:25
6	Introduction to sociology [6]	14:58
7	History of Fishes I Lecture [7]	14:58
8	Word Vector Representation [8]	1:18:16
9	How bitcoin achieves decentralization [9]	1:13:40

Table 1: Description of videos used in the pipeline evaluation

6.1 Stage 1: Shot Boundary Detection

We measure the performance of the shot boundary detection algorithm by comparing against manually identified ground-truth boundaries in the nine videos. Results are presented in Table 2.

The average F1-Score across the nine videos was 0.75. Video #9 had a particularly low F1-score (0.39) because it frequently alternates between the instructor headshot, the instructor alongside the slide view, and the fullscreen slide view. Because our algorithm detects shots only when text and image change together, it misses cases where only one of them changes (e.g., when a same slide is shown but the screen is just zoomed in).

ID	Num. Shots	Precision	Recall	F1-Score
1	17	0.77	1	0.87
2	13	0.87	0.68	0.73
3	3	0.60	0.60	0.60
4	8	0.71	0.62	0.67
5	29	0.81	0.93	0.87
6	7	1	0.86	0.92
7	26	1	0.90	0.95
8	76	0.63	0.89	0.74
9	40	0.64	0.28	0.39

Table 2: Summary of evaluation of accuracy in shot boundary detection

[1] https://memento.epfl.ch/event/mooc-a-resilient-future-science-and-technology-for
[2] https://www.youtube.com/watch?v=ovC98yvRZe8
[3] https://www.youtube.com/watch?v=hu7iGGnzq3Y
[4] https://www.youtube.com/watch?v=xrFtN_UJhYc
[5] https://www.edx.org/course/journalism-social-change-uc-berkeleyx-j4sc101x-1
[6] https://www.youtube.com/watch?v=dZv3a14weDA
[7] https://www.youtube.com/watch?v=23mNQZ6uhyI
[8] https://www.youtube.com/watch?v=ERibwqs9p38
[9] https://www.youtube.com/watch?v=q5GWwTgRIT4

6.2 Stage 2: Text Segmentation within Slides

We measure the performance of the text segmentation algorithm by comparing against manually identified text segments in the videos. Results are presented in Table 3. Correct refers to the number of text segments generated by the algorithm that exist in the ground truth text segmentation set.

The average recall across the nine videos was 0.67. Videos with low recall (videos #3: recall=0.44 and #4: recall=0.38) have unusually big font sizes, and our algorithm misses certain groupings because the fixed parameters in our algorithm are sensitive to font size when analyzing the layout and relative spacing between items.

ID	Num. Groups	Correct	Recall
1	69	55	0.80
2	74	64	0.86
3	18	8	0.44
4	21	8	0.38
5	41	33	0.80
6	15	9	0.60
7	125	95	0.76
8	241	167	0.69
9	244	181	0.74

Table 3: Summary of evaluation of accuracy in text segmentation (Num. Groups is number of groups in the ground truth dataset)

6.3 Stage 3: Text-to-Script Alignment

We measure the performance of the text-to-script alignment algorithm by comparing against manually found references in the videos. We chose five of the videos from our set (ID 1-5), which include 141 references from the ground truth reference set. One of the authors constructed the ground truth reference set by manually annotating all references from the five videos.

The average F1-Score across the five videos was 0.79. Videos with high F1-Score (videos #1: F1=0.89 and #5: F1=0.87) have a relatively simple structure, in which the instructor explains each item on the slide in order using a single sentence. In contrast, in video #4 (F1=0.67), the same concept recurs repeatedly across the script, which makes it difficult to find references accurately.

ID	Num. Gold References	Precision	Recall	F1-Score
1	33	0.84	0.94	0.89
2	57	0.78	0.68	0.73
3	14	0.75	0.86	0.80
4	8	0.71	0.62	0.67
5	29	0.81	0.93	0.87

Table 4: Summary of evaluation of accuracy in reference extraction

7 PILOT STUDY

We conducted a user study to assess how DynamicSlide's reference-based interaction techniques help learning with slide-based videos. For comparison, we built a baseline player without the reference-based techniques but with thumbnail of slides and transcript panel. User can navigate to the starting point of a slide or a line of transcript by clicking it, inspired by other research prototypes for lecture videos [19, 41]. Our study was designed to answer the following research questions:

- **RQ1.** Can DynamicSlide's highlighting feature lessen the cognitive demand of watching lecture videos compared to the baseline player?
- **RQ2.** Does DynamicSlide's item-based navigation facilitate information search?
- **RQ3.** How do users use DynamicSlide's item-based note-taking feature?

7.1 Participants

Participants were recruited from a university via an online advertisement on a community website. We recruited 12 participants for the study (mean age=25.5, stdev=3.89, max=34, min=21). Out of 12 participants, four were female and four were graduate students (the rest were undergraduate students). All participants had prior experience watching lecture videos online. All of them previously took an introduction to programming course, and had the prerequisite knowledge for the lectures used in study. Each session lasted about 50-minutes, and participants received $8 compensation.

7.2 Study Design

The study used a within-subjects design. Participants watched one video with the baseline player, and another video with DynamicSlide. We counterbalanced the order of the interfaces and the videos presented.

We selected two videos from a series of lecture "Introduction to Data Mining". The first video [12] was 4:51 long and contained three slides, while the second video [11] was 3:46 long with two slides.

7.3 Procedure

Before watching each video, we gave a 5-minute tutorial for the video player interface. After watching the video, participants completed the NASA-TLX questionnaire [16], which we used to measure the cognitive load involved in watching the video. Then, participants were given five information search tasks, in which they were asked to locate the part of the video that mentioned the given information. There were three types of information: (1) information that appears only in the slide (Slide Only), (2) information that is mentioned in both the slides and the scripts (Slide & Script), and (3) information that is mentioned only in the scripts (Script Only). We recorded the completion time for each task, as well as the usage logs of each feature in the UI. After finishing the search tasks, participants evaluated the usefulness and ease of use of the provided interface features in a questionnaire based on Davis et al. [9]. They repeated the process for the other interface. After watching both videos, participants evaluated the helpfulness of the three reference-based features in DynamicSlide's player. The session ended with a semi-structured interview about the participants' opinion on the automatic highlighting feature, and their preferences for highlighting styles. Participants were also asked about possible use cases of the note-taking feature.

7.4 Results

We report findings on how participants used the reference-based interaction techniques in learning tasks.

7.4.1 RQ1. Highlighting lessens the cognitive demand of watching lecture videos. Users prefer "Pointing" or "Blur" to "Hide" effect for highlighting. Overall participants reported less cognitive load using DynamicSlide compared to the baseline (DynamicSlide/baseline – overall: 3.1/3.2, mental: 3.2/3.5, temporal: 3.0/3.0, performance: 2.7/3.3, effort: 3.5/3.2, frustration: 2.6/3.1), although the differences were not statistically significant. 11 out of 12 participants responded that the highlighting feature was helpful. Most of them noted the benefit of being able to return to the correct place in the slide after distractions. For example, U8 mentioned *"I easily get distracted in a classroom or when taking online courses. The red dot on the video player helped me get back to the lecture."* U4 noted *"Since I'm not good at English, I have to go back and forth between the subtitle and the slide. The red dot makes it much easier to go back to the slide."* These participants used the pointer as an external aid to stay focused and relate the slide content to the narration easily. Meanwhile, U12, who did not find the feature helpful, commented that *"I like to look at the entire slide to grasp the overall content. The red dot takes away my attention from the big picture."*

Among the three different styles of highlighting, most participants preferred "pointing" and "blur" over "hide". The main reason for the preference was because "pointing" and "blur" effects preserve the overall content of the slide, whereas the "hide" effect masks part of the slide. Users felt they were deprived of existing content. This may be due to the fact that users knew there was masked content underneath (before they turned on the feature). On the other hand, it could point to a practical design implication for authoring slides. For example, using an effect similar to "pointing" or "blur" may be more effective than revealing bullet points sequentially (analogous to the "hide" effect).

7.4.2 RQ2. Item-based navigation facilitates searching information in slides. Users found information faster using DynamicSlide compared to the baseline, when the information was included in the slide (DynamicSlide/baseline – slide only: 18.8/ 35.5, slide script: 15.3/20.5, in seconds). As expected, item-based navigation did not help when the information was only in the script (DynamicSlide/baseline : 32.6/27.7). The performance between users varied widely and the results were not statistically significant. Most participants used slide-based navigation as a first, macro step for search, and then searched for the details using the slide items or the subtitle panel.

Overall, participants found the item-based navigation feature useful (5.9 out of 7). In addition to the search task, several participants commented that the fine-grained, non-linear control over the video could be useful for skimming or reviewing lecture videos.

7.4.3 RQ3. Item-based note-taking is useful for reviewing longer videos. While participants rated the item-based note-taking feature

useful (4.3 out of 7), they mentioned that it would be more useful for watching a longer video or for reviewing a lecture. Participants mentioned they would use the note-taking feature to mark important or confusing points and revisit them later. One participant noted, *"When I am taking a lecture, I build my own knowledge model for the subject and decide how important each part of the lecture is. In that case, the note-taking feature will be useful. But in this experimental setting I needed more time to get familiar with the topic in order to take notes."* Another user responded, *"It will be useful when preparing for an exam, since I don't have enough time to watch the whole video again."*

8 DISCUSSION

In this section, we discuss the implications and limitations of the current techniques and evaluation. Then we outline some directions for future work.

8.1 Cost of Incorrect Reference Estimation

Because our reference estimation algorithm is not perfect, the cost of incorrect estimation has to be carefully understood. Incorrect references may disturb the video learning experience, as highlighting or navigation to incorrect points will confuse users. Applying more advanced methods for measuring similarity between text, or leveraging crowdsourcing can potentially improve the current pipeline for finding references. Different interaction techniques are affected differently by incorrect reference estimation. For example, in DynamicSlide's emphasis feature, the "hide" option is more fragile to incorrect reference estimation compared to other options, because it hides content that might be relevant whereas in other options it is still visible on screen. Providing the user with control over emphasis options can reduce the damage caused by incorrect reference estimation, while giving them the flexibility to see the slide in their preferred way.

8.2 Beyond Text-Based Slides

The technique and prototype of the system suggested in this paper was demonstrated with slide-based videos consisting mostly of text. However, the idea and approach can be applied to other informational videos that contain multiple types of objects (e.g., figures, charts, and tables) on a single slide by expanding the definition of *reference* to cover a pair of any content object in a slide and the corresponding segment of narration. With computer vision and natural language processing techniques, we will be able to infer the relationship between the instructor's explanation and non-text objects in the slide and leverage these references for interaction.

8.3 Editing Support for Slide-Based Lecture Video

The technique for automatically identifying references can be further used to reduce the time spent in editing existing slide-based lecture videos. For example, instructors can easily edit or emphasize the word in a slide by rendering new text over the original text, without having to re-record the video. Instructors can easily add highlighting effects on their video by assigning references between an item in the slide and the corresponding segment of narration. The discretized representation of references can make it easier to update an item in the slide with the corresponding narration, without having to edit the entire video or the source presentation file. This can dramatically reduce the time and expertise required to edit videos.

8.4 Limitations

Even though participants gave positive comments on using reference-based interaction techniques for learning tasks, the one-time study session does not accurately simulate realistic learning scenarios, which often involve longer videos and re-watching for working on assignments and preparing for exams. Also, our participants were not native English speakers, which made them put significant effort into reading subtitles and probably affected the difficulty and completion time of the tasks. Finally, our evaluation mostly consisted of self-reported measures of cognitive load and usability. A long-term evaluation with realistic learning tasks and other measures of cognitive load such as eye tracking will provide a deeper understanding of the effectiveness of our interaction techniques.

9 CONCLUSION

To improve the learner's experience of slide-based lecture videos, this paper focuses on the concept of reference, the relationship between objects in a slide and the instructor's verbal narration. References have the potential to enable effective video interaction as they provide a discretized representation of the video. We present DynamicSlide a prototype video processing and browsing system that automatically extracts references from existing slide-based lecture videos, and supports automatic highlighting, item-based navigation, and note-taking interactions. Our evaluation shows that references can be found automatically for a variety of slide-based videos and can be used to improve the video learning experience. Furthermore, reference-based video interaction opens up unique opportunities for connecting visual and spoken contents in video and enabling easier watching, navigation, and editing of educational contents.

10 ACKNOWLEDGMENTS

The authors thank John Joon Young Chung for his support in running experiment and members of KIXLAB at KAIST for their support and feedback. This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korean government (MSIT) (No.2017-0-01217, Korean Language CALL Platform Using Automatic Speech-writing Evaluation and Chatbot).

REFERENCES

[1] Don Adjeroh, M C Lee, N Banda, and Uma Kandaswamy. 2009. Adaptive Edge-Oriented Shot Boundary Detection. *EURASIP Journal on Image and Video Processing* 2009 (2009), 1–13. https://doi.org/10.1155/2009/859371

[2] Reza Asadi, Harriet J. Fell, Timothy Bickmore, and Ha Trinh. 2016. Real-time presentation tracking using semantic keyword spotting. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH* 08-12-Sept (2016), 3081–3085. https://doi.org/10.21437/Interspeech.2016-617

[3] Nikola Banovic, Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2012. Waken. In *Proceedings of the 25th annual ACM symposium on User interface software and technology - UIST '12.* ACM Press, New York, New York, USA, 83. https://doi.org/10.1145/2380116.2380129

[4] Arijit Biswas, Ankit Gandhi, and Om Deshmukh. 2015. MMToC: A Multimodal Method for Table of Content Creation in Educational Videos. In *Proceedings of*

the 23rd ACM international conference on Multimedia - MM '15. ACM Press, New York, New York, USA, 621–630. https://doi.org/10.1145/2733373.2806253

[5] Bitcoin and Cryptocurrency Technologies Online Course. 2015. Lecture 1 - Intro to Crypto and Cryptocurrencies. Video. Retrieved August 16, 2018 from https://www.youtube.com/watch?v=fOMVZXLjKYo.

[6] Xiaoyin Che, Haojin Yang, and Christoph Meinel. 2013. Lecture video segmentation by automatically analyzing the synchronized slides. *Proceedings of the 21st ACM international conference on Multimedia - MM '13* (2013), 345–348. https://doi.org/10.1145/2502081.2508115

[7] Konstantinos Chorianopoulos. 2018. A Taxonomy of Asynchronous Instructional Video Styles. *The International Review of Research in Open and Distributed Learning* 19, 1 (feb 2018). https://doi.org/10.19173/irrodl.v19i1.2920

[8] Andrew Cross, Mydhili Bayyapunedi, Dilip Ravindran, Edward Cutrell, and William Thies. 2014. VidWiki. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing - CSCW '14*. ACM Press, New York, New York, USA, 1167–1175. https://doi.org/10.1145/2531602.2531670

[9] Fred D. Davis. 1989. Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly* 13, 3 (sep 1989), 319. https://doi.org/10.2307/249008 arXiv:arXiv:1011.1669v3

[10] Laurent Denoue, Scott Carter, Matthew Cooper, and John Adcock. 2013. Real-time direct manipulation of screen-based videos. *Proceedings of the companion publication of the 2013 international conference on Intelligent user interfaces companion - IUI '13 Companion* (2013), 43. https://doi.org/10.1145/2451176.2451190

[11] Data Science Dojo. 2017. Intro to Data Mining: Data Attributes(Part2). Video. Retrieved August 9, 2018 from https://www.youtube.com/watch?v=xrFtN_UJhYc.

[12] Data Science Dojo. 2017. Introduction to Data Mining: Data Attributes (Part 1). Video. Retrieved August 9, 2018 from https://www.youtube.com/watch?v=hu7iGGnzq3Y.

[13] FFmpeg. 2018. FFmpeg. http://www.ffmpeg.org.

[14] Elena L. Glassman, Juho Kim, Andrés Monroy-Hernández, and Meredith Ringel Morris. 2015. Mudslide. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*. ACM Press, New York, New York, USA, 1555–1564. https://doi.org/10.1145/2702123.2702304

[15] Google. 2018. Google Cloud Vision API. https://cloud.google.com/vision.

[16] Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. *Advances in Psychology* 52, C (1988), 139–183. https://doi.org/10.1016/S0166-4115(08)62386-9 arXiv:arXiv:1011.1669v3

[17] Chen-Tai Kao, Yen-Ting Liu, and Alexander Hsu. 2014. Speeda: Adaptive Speed-up for Lecture Videos. In *Proceedings of the adjunct publication of the 27th annual ACM symposium on User interface software and technology - UIST'14 Adjunct*. ACM Press, New York, New York, USA, 97–98. https://doi.org/10.1145/2658779.2658794

[18] Kandarp Khandwala and Philip J Guo. 2018. Codemotion. In *Proceedings of the Fifth Annual ACM Conference on Learning at Scale - L@S '18*. ACM Press, New York, New York, USA, 1–10. https://doi.org/10.1145/3231644.3231652

[19] 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 - UIST '14*. ACM Press, New York, New York, USA, 563–572. https://doi.org/10.1145/2642918.2647389

[20] Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, Vol. 10. 707–710.

[21] Ching Liu, Juho Kim, and Hao-chuan Wang. 2018. ConceptScape. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*. ACM Press, New York, New York, USA, 1–12. https://doi.org/10.1145/3173574.3173961

[22] Han Lu, Sheng Syun Shen, Sz Rung Shiang, Hung Yi Lee, and Lin Shan Lee. 2014. Alignment of spoken utterances with slide content for easier learning with recorded lectures using structured support vector machine (SVM). In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*. 1473–1477.

[23] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[24] Toni-Jan Keith Palma Monserrat, Shengdong Zhao, Kevin McGee, and Anshul Vikram Pandey. 2013. NoteVideo. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*. ACM Press, New York, New York, USA, 1139. https://doi.org/10.1145/2470654.2466147

[25] 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 - CHI '15*. ACM Press, New York, New York, USA, 1565–1568. https://doi.org/10.1145/2702123.2702209

[26] Cuong Nguyen and Feng Liu. 2016. Gaze-based Notetaking for Learning from Lecture Videos. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*. ACM Press, New York, New York, USA, 2093–2097. https://doi.org/10.1145/2858036.2858137

[27] Robert M Ochshorn and Max Hawkins. 2018. Gentle. https://lowerquality.com/gentle/.

[28] OpenCV. 2018. OpenCV. https://opencv.org/.

[29] Amy Pavel, Colorado Reed, Björn Hartmann, and Maneesh Agrawala. 2014. Video digests. In *Proceedings of the 27th annual ACM symposium on User interface software and technology - UIST '14*. ACM Press, New York, New York, USA, 573–582. https://doi.org/10.1145/2642918.2647400

[30] Luca Ponzanelli, Gabriele Bavota, Andrea Mocci, Massimiliano Di Penta, Rocco Oliveto, Mir Hasan, Barbara Russo, Sonia Haiduc, and Michele Lanza. 2016. Too long; didn't watch!. In *Proceedings of the 38th International Conference on Software Engineering - ICSE '16*. ACM Press, New York, New York, USA, 261–272. https://doi.org/10.1145/2884781.2884824

[31] Jasmine Rana, Henrike Besche, and Barbara Cockrill. 2017. Twelve tips for the production of digital chalk-talk videos. *Medical Teacher* 39, 6 (jun 2017), 653–659. https://doi.org/10.1080/0142159X.2017.1302081

[32] Hijung Valentina Shin, Floraine Berthouzoz, Wilmot Li, and Frédo Durand. 2015. Visual transcripts. *ACM Transactions on Graphics* 34, 6 (oct 2015), 1–10. https://doi.org/10.1145/2816795.2818123

[33] Tafuto. 2012. An Introduction to Political Science Through Classic Political Works. Retrieved August 16, 2018 from https://www.youtube.com/watch?v=Xhjl-MvnqAc.

[34] Ottokar Tilk and Tanel Alumäe. 2015. LSTM for Punctuation Restoration in Speech Transcripts. In *Interspeech 2015*. Dresden, Germany.

[35] Shoko Tsujimura, Kazumasa Yamamoto, and Seiichi Nakagawa. 2017. Automatic Explanation Spot Estimation Method Targeted at Text and Figures in Lecture Slides. In *Interspeech 2017*, Vol. 2017-Augus. ISCA, ISCA, 2764–2768. https://doi.org/10.21437/Interspeech.2017-750

[36] Xiang Xiao and Jingtao Wang. 2017. Undertanding and Detecting Divided Attention in Mobile MOOC Learning. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*. ACM Press, New York, New York, USA, 2411–2415. https://doi.org/10.1145/3025453.3025552

[37] Kuldeep Yadav, Ankit Gandhi, Arijit Biswas, Kundan Shrivastava, Saurabh Srivastava, and Om Deshmukh. 2016. ViZig: Anchor Points based Navigation and Summarization in Educational Videos. In *Proceedings of the 21st International Conference on Intelligent User Interfaces - IUI '16*. ACM Press, New York, New York, USA, 407–418. https://doi.org/10.1145/2856767.2856788

[38] Shir Yadid and Eran Yahav. 2016. Extracting code from programming tutorial videos. In *Proceedings of the 2016 ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software - Onward! 2016*. ACM Press, New York, New York, USA, 98–111. https://doi.org/10.1145/2986012.2986021

[39] Natsuo Yamamoto, Jun Ogata, and Yasuo Ariki. 2003. Topic Segmentation and Retrieval System for Lecture Videos Based on Spontaneous Speech Recognition. In *Proceedings of the 8th European Conference on Speech Communication and Technology - EUROSPEECH*. 961–964.

[40] Haojin Yang and Christoph Meinel. 2014. Content Based Lecture Video Retrieval Using Speech and Video Text Information. *IEEE Transactions on Learning Technologies* 7, 2 (apr 2014), 142–154. https://doi.org/10.1109/TLT.2014.2307305

[41] Baoquan Zhao, Shujin Lin, Xiaonan Luo, Songhua Xu, and Ruomei Wang. 2017. A Novel System for Visual Navigation of Educational Videos Using Multimodal Cues. In *Proceedings of the 2017 ACM on Multimedia Conference - MM '17*. ACM Press, New York, New York, USA, 1680–1688. https://doi.org/10.1145/3123266.3123406