SoftVideo: Improving the Learning Experience of Software Tutorial Videos with Collective Interaction Data

Saelyne Yang
saelyne@kaist.ac.kr
School of Computing, KAIST
Daejeon, Republic of Korea

Jisu Yim
yimjisu99@kaist.ac.kr
School of Computing, KAIST
Daejeon, Republic of Korea

Aitolkyn Baigutanova
aitolkyn.b@kaist.ac.kr
School of Computing, KAIST
Daejeon, Republic of Korea

Seoyoung Kim
youthskim@kaist.ac.kr
School of Computing, KAIST
Daejeon, Republic of Korea

Minsuk Chang
minsuk.chang@navercorp.com
NAVER AI LAB
Seongnam, Republic of Korea

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

ABSTRACT

Many people rely on tutorial videos when learning to perform tasks using complex software. Watching the video for instructions and applying them to target software requires frequent going back-and-forth between the two, which incurs cognitive overhead. Furthermore, users need to constantly compare the two to see if they are following correctly, as they are prone to missing out on subtle differences. We propose SoftVideo, a prototype system that helps users plan ahead before watching each step in tutorial videos and provides feedback and help to users on their progress. SoftVideo is powered by collective interaction data, as experiences of previous learners with the same goal can provide insights into how they learned from the tutorial. By identifying the difficulty and relatedness of each step from the interaction logs, SoftVideo provides information on each step such as its estimated difficulty, lets users know if they completed or missed a step, and suggests tips such as relevant steps when it detects users struggling. To enable such a data-driven system, we collected and analyzed video interaction logs and the associated Photoshop usage logs for two tutorial videos from 120 users. We then defined six metrics that portray the difficulty of each step, including the time taken to complete a step and the number of pauses in a step, which were also used to detect users' struggling moments by comparing their progress to the collected data. To investigate the feasibility and usefulness of SoftVideo, we ran a user study with 30 participants where they performed a Photoshop task by following along a tutorial video with SoftVideo. Results show that participants could proactively and effectively plan their pauses and playback speed, and adjust their concentration level. They were also able to identify and recover from errors with the help SoftVideo provides.

IUI '22, March 22–25, 2022, Helsinki, Finland

ACM ISBN 978-1-4503-9144-3/22/03…$15.00
https://doi.org/10.1145/3490099.3511106

CCS CONCEPTS

• **Human-centered computing** → **Interactive systems and tools**.

KEYWORDS

video interaction, software tutorial videos, interaction log analysis, data-driven interface

ACM Reference Format:
Saelyne Yang, Jisu Yim, Aitolkyn Baigutanova, Seoyoung Kim, Minsuk Chang, and Juho Kim. 2022. SoftVideo: Improving the Learning Experience of Software Tutorial Videos with Collective Interaction Data. In *27th International Conference on Intelligent User Interfaces (IUI '22), March 22–25, 2022, Helsinki, Finland.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3490099.3511106

1 INTRODUCTION

Tutorial videos provide step-by-step instructions of complex tasks for feature-rich software such as Photoshop [45] and AutoCAD [3]. People watch a tutorial video and try to apply the techniques from the video to their software when learning new techniques [25]. For example, they search for a video about "removing background from an image" and learn the skill by applying it to their own image.

When following a tutorial video, people often watch instructions and apply them to their own work (e.g., image editing, document editing, video authoring, programming, etc.) by alternating between the video and the software. Commonly, they first watch a step in the video and apply it to their application. If the application results an error or an unintended outcome, users often adjust the pace of the video and rewatch the step, trying to find what they did differently. Most people go through multiple trial-and-error cycles, which could be cumbersome.

Also, when applying instructions from a tutorial video to their software, users need to constantly compare the two to see if they are following correctly. Users can easily miss important details when a demonstration in a video moves too quickly [25], or subtle visual changes are presented in the video [25, 53]. This process is cognitively demanding with constant context switching and is prone to mistakes.

In this research, we propose SoftVideo, a prototype system that helps users plan ahead before watching each step in tutorial videos, gives feedback to users on their progress, and provides help to overcome confusing moments. Users can see step information such

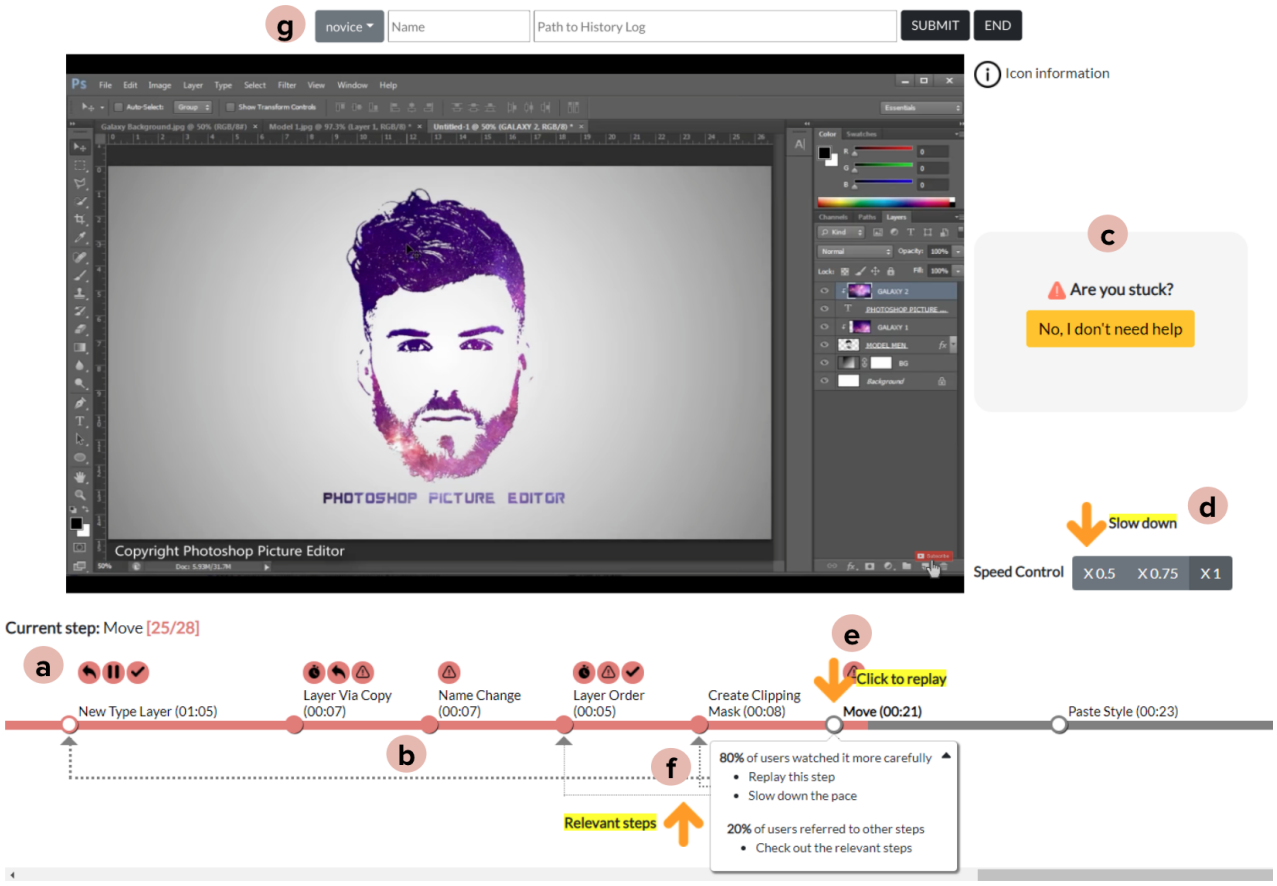


Figure 1: Overview of SoftVideo. Along with the software tutorial video, SoftVideo provides (a) a timeline where users can see the action name, its length, and the estimated difficulty. (b) Users can receive real-time feedback on their progress. If a user followed a step, the circle will be filled. (c) SoftVideo detects users' confusing moments. Once detected, it provides users with suggestions such as (d) slowing down the pace, (e) replaying the step, or (f) seeing relevant steps. Users see customized information based on (g) the expertise level they enter.

as the name of an action or the duration and difficulty of each step to anticipate what is upcoming and prepare, which reduces context-switching overhead. Users also get informed about whether they completed a step or not so that they can be aware of any missed steps. Lastly, users struggling at a particular step can get help suggestions such as slowing down the pace, replaying the step, or seeing relevant steps. SoftVideo detects users' confusing moments automatically and presents help suggestions at appropriate moments.

To build SoftVideo, we leverage previous learners who had watched the same tutorial and worked toward the same end goal. Collective interaction logs of the video and the software from previous learners can reveal patterns of how people learn from the tutorial. For example, analyzing the logs can detect the steps people frequently struggle in or miss. It can also identify when the user is facing difficulties by comparing their progress to previous learners. Furthermore, it can reveal how people overcome confusing moments, such as by looking at which steps they referred to when completing a step.

We chose Adobe Photoshop as an instance of the software. We collected interaction logs composed of video interactions (i.e., pause, play, jump) in synchronization with Photoshop usages (i.e., actions performed in the software). Collecting interaction data of both sources in a synchronized manner is essential as it captures the actual interaction between the two sources. This allows for more accurate estimations of the user's current task state, enabling SoftVideo to provide appropriate help to people facing the back-and-forth challenges.

We collected 120 complete interaction logs with two tutorial videos (60 logs for each) with 74 participants of varying levels of expertise in Photoshop. Our data analysis pipeline then analyzed the collected data to 1) estimate the difficulty of each step by analyzing how users behaved on each step and 2) identify the relevancy of each step. For 1), we define six measures that portray the difficulty of each step: Execution Time Index, Repetition Time Index, Backjump Frequency, Pause Frequency, Miss Rate, and Re-follow Rate. For 2), we identify the "Relevant steps" of each step, which are the steps that are performed again in order to complete a particular step.

We evaluated our tool with the two Photoshop tutorial videos with which we collected interaction data. We recruited 30 participants (23 novices, 7 experienced) and asked them to follow a tutorial video with SoftVideo. Results show that participants were able to proactively and effectively plan their pauses and playback speed, and vary their concentration level before watching a step by looking at the presented step information. The difficulty visualization also made them feel relieved when they encountered confusing moments. They were also able to identify and recover from errors with the help SoftVideo provided. Relevant step information helped them overcome confusing moments and acquire contextual Photoshop knowledge.

The primary contributions of this paper are as follows:

- A publicly available dataset of 120 interaction logs across the tutorial videos and Photoshop in use [1].
- SoftVideo, an interface powered by previous users' interaction data that provides step information and real-time feedback to users.
- Results from a study showing that participants used the system to efficiently plan their action and recover from errors in Photoshop tasks.

2 RELATED WORK

Users of complex software often face challenges when performing a task, such as understanding functionalities in software [20]. They need additional help to use software [25], which applies even to professionals [24]. Video demonstrations of software have been a popular source of help for learners, as they provide detailed visual descriptions [25, 46]. Our work improves the learning experience of software tutorial videos by providing appropriate information and help, powered by the analysis of the interaction data of when people apply the tutorial video to their software in use. We review related work on analyzing video and software interaction data, improving software tutorials, and providing real-time suggestions.

2.1 Analyzing Video Interaction Data

A stream of research has analyzed interaction data of educational videos to gain insights into learners' understanding of the video. A number of work analyzed interaction sequences to relate with learners' engagement and performance [5, 5, 21, 30, 31, 35, 48]. Another stream of research has analyzed video interaction data to reveal meaningful insights of the videos such as perceived difficulty [36] or important moments of the video [11, 26]. Kim et al. [27] have analyzed dropouts and peaks of interactions in different types of videos and suggested design implications for better video learning experiences. Li et al. [35] have analyzed in-video interactions together with a survey about perceived video difficulty to find relevant video interactions that indicate a student has experienced difficulty. However, little research has explored synchronized interaction data with both the video and the corresponding application where users watch a video to complete a task. It is difficult to fully estimate a users' state with only video interaction data. For example, even if a user watched the whole video, it is difficult to know how well the user could follow the content while watching it. Therefore, in this research, we aim to identify meaningful information of videos such as in-step difficulties and relevant steps by using the synchronized interaction data.

[1] softvideo.kixlab.org

2.2 Analyzing Software Usage Logs

Collective software usage logs can reveal meaningful insights into how people use the software. A stream of research used applications' logs to identify frequent tasks [14] or recommend workflows by comparing multiple workflows [7, 41, 49]. Another stream of research classified a sequence of commands to provide an overview of command sequences and thereby support semantic navigation [8, 13, 38]. Others approached to integrate the analysis of software usage logs into the software interface or software tutorial videos. Patina [40] visualizes collective usage patterns of UI elements on the software interface to help users use the software efficiently. Fraser et al. [17] used the software usage logs to further assist segmenting software tutorial videos into meaningful sections. Extending the previous work, we analyze software usage logs to reveal meaningful information of corresponding tutorial videos such as step difficulties and relevancy.

2.3 Improving Software Tutorials

For creating better software tutorials, researchers have introduced ways to easily create contextual tutorials for learning GUI applications [34, 54] and to capture tutorials from the user demonstrations on the software [9, 19, 32, 33]. Focusing on tutorial videos, there have been efforts to improve the use of tutorials, such as contextually presenting appropriate videos to reduce loads of navigation [18], segmenting the tutorial videos into meaningful units [17, 28, 47], and extracting information from the video such as time-series interaction data [37] and UI elements [4]. Some techniques have focused on supporting the context switching process between the tutorial video and the target software, such as automatically controlling the video playback depending on users' progress on the software [46] or integrating the software context into the video, allowing users to learn from the video without switching to the software [43]. We extend the previous work by presenting data-driven information on tutorial videos so that learners can effectively manage their context switching behavior and recover from errors.

2.4 Providing Real-time Suggestions for Better User Experience

A thread of research aims to provide real-time suggestions to guide the user in various applications. ViZig [52] and LectureScape [26] help learners watching educational videos easily navigate to where they want by providing anchor points. WebICLite [55] recommends relevant web pages for users to look at to help web surfing. Furthermore, user interfaces that adapt to user data such as Adaptive Hypermedia [6] or Ephemeral Adaptation [16] can improve the user experience. Patina [40] automatically displays new UI components when users navigate through software features. In this research, we aim to provide user-adaptive real-time suggestions to help learners follow software tutorial videos by detecting confusing moments and suggesting relevant parts in the video.

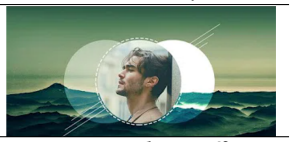
	1) Logo	2) Geometry
Outcome		
Effect	Galaxy-style logo design	Geometric Shape Effect
Length	9m 35s	7m 34s
Number of Actions	27	45
URL	youtu.be/ifG1SDxqpAQ	youtu.be/vcLjyGbF40Y

Table 1: Tutorial videos used in the data collection study.

3 DATA COLLECTION STUDY

In our approach, we leverage interaction logs from previous learners who had watched the same tutorial and worked toward the same end goal. Collective interaction logs of video and the software can provide useful insights into patterns of how people learn from the tutorial. It can reveal meaningful information of videos, such as where users struggle a lot and thus need to pay attention to. We recruited participants to collect interaction data of both the tutorial video and the software in synchronization. We used Adobe Photoshop as the target software, due to its high availability and popularity. Participants were asked to follow Photoshop tutorial videos and complete image editing tasks.

3.1 System for Data Collection

We built a system to collect the interaction data from both the tutorial video and Photoshop synchronously (Figure 2). The system collects video interaction logs (i.e., play, pause, and jump actions with the corresponding video timestamp and user timestamp) in synchronization with software interaction logs (i.e., actions done in Photoshop). In the system, we embedded a Youtube video player for a Photoshop tutorial video. We logged video interaction data using the YouTube player API [2]. To log software interaction logs, we used the History Log feature available in Photoshop. Once users enable the History Log feature in Photoshop, a text file that logs the action history is saved in their local computer. A new line is appended to the file for every action performed in Photoshop. Once a user uploads the path of the text file to our system in the beginning, the system reads the changes in the file periodically and logs the actions in Photoshop, together with the corresponding video timestamp and user timestamp. We stored the logs in Firebase Realtime Database [12].

3.2 Participants

We recruited 75 participants from an academic institution through online recruitment postings (48 male, 27 female, mean age 23). We collected their frequency of Photoshop usage on a 5-point scale (1: None, 2: Yearly, 3: Monthly–Yearly, 4: Monthly, 5: Weekly). Based on their responses, we grouped participants who have not used Photoshop or use it 1-2 times a year as novice, and experienced otherwise. We used the frequency of use for grouping expertise because new features are added to the software several times a year [22] and to avoid subjective measures (e.g., self-reported expertise). Each participant completed either one or two tutorials depending on their availability during the given time. The number of collected logs for each tutorial and participants' expertise level is shown in Table 2. Participants were compensated with 20,000 KRW (approximately 17 USD) for a 90-minute-long study.

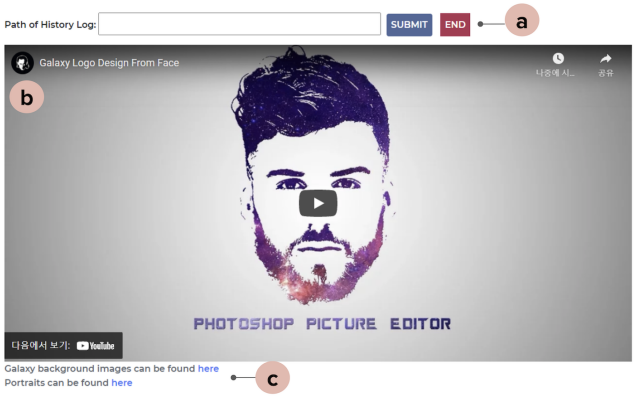


Figure 2: The system used in the data collection study. To collect interaction logs, (a) the path to the History Log file should be uploaded to the system in the beginning, after enabling the History Log feature in Photoshop. (b) The tutorial video. (c) We provided participants with links to images that they can use for the tasks.

3.3 Task

The task was to follow a Photoshop tutorial video about making 1) a galaxy-style logo design ('Logo') or 2) a geometric shape effect ('Geometry') (Table 1). We chose the videos from YouTube because they were less than 10 minutes to ensure a feasible study duration, and the tasks were not too trivial (e.g., image cropping) nor too advanced (e.g., poster design).

3.4 Procedure

Participants were first assigned to one of the two tutorials. After we introduced the effect and the final outcome of the tutorial, they were asked to prepare images they wanted to use. Participants could optionally choose one of the images we provided. They were

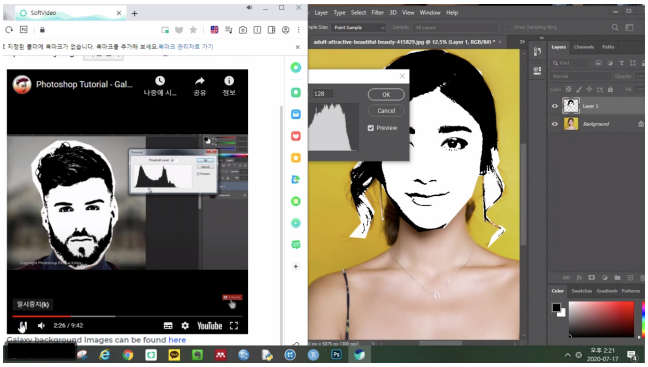


Figure 3: An example session of the data collection study. A participant is following the tutorial video (on the left) on their software (on the right).

then instructed to open Photoshop and our system, and follow the tutorial video. If time allowed after completing one, they followed another tutorial.

The study was conducted in either an offline or online setting. The same system was used in both settings.

- Offline setting: We set up computers with Photoshop installed. We enabled the Photoshop History Log feature and uploaded the path of the log file to our system. A total of 24 participants joined offline.
- Online setting: Participants were asked to install Photoshop and either Whale [51] or Min web browsers [50] before the study to enable real-time tracking of Photoshop usage logs, as other browsers did not support it due to their security policies. They were asked to enable screen sharing during the study. We guided them to enable the Photoshop History Log feature and upload the path of the log file to our system. A total of 51 participants joined online.

3.5 Results

With 75 participants, we collected a total of 120 interaction data, 60 for each of the tutorials (Table 2). The interaction data is composed of *video interaction logs* and *software usage logs*. Below we specify the scope of the video interaction logs and the software usage logs we collected.

- Video interaction logs: Play, Pause (duration) and Jump (from, to) on the video and the corresponding user timestamps and video timestamps.
- Software usage logs: Actions done on the software (e.g., `Crop`, `Resize`) and the corresponding user timestamps and video timestamps.

The average time taken to complete the tutorial was 32m 54s and 29m 35s for the Logo and Geometry tutorials, respectively (Table 3).

	Novice (N=59)	Experienced (N=16)
1) Logo	49	11
2) Geometry	48	12

Table 2: The number of collected logs for each tutorial.

	Novice	Experienced	Avg.
1) Logo	35m 24s	21m 46s	32m 54s
2) Geometry	30m 51s	24m 33s	29m 35s

Table 3: Average time taken to complete each tutorial.

4 DATA ANALYSIS PIPELINE

Our data analysis pipeline analyzes the collected interaction data to identify meaningful information from the tutorial video. Specifically, we aim to 1) estimate the difficulty of each step so that users can plan their action before watching each step, and 2) identify the relatedness of steps so that users can refer to when having difficulties in a particular step. We first describe measures that are used for each of the two purposes.

4.1 Measures

4.1.1 Difficulty of steps. We defined six measures that portray the difficulty of each step: Execution Time, Repetition Index, Backjump Frequency, Pause Frequency, Miss Rate, and Re-follow Rate. Table 4 shows the definitions of six measures. Below we describe each measure in detail.

- **Execution Time Index**: *(Time taken to follow a step)/(video time)*. If a user spends much longer time in a certain step than its length in the video, there is a high chance that the user has difficulties completing the step. For a fair comparison between the steps, we take relative execution time, defined as the time taken to follow a step divided by the video length of the corresponding step. Note that there was no fast-winded or cut parts in the videos we used.
- **Repetition Time Index**: *(Total time of a step being watched)/(video time)*. Users repeatedly watch a step if something is unclear from the video or does not work in their context. Similar to Execution Time Index, we take relative repetition time, defined as the total time of a step being watched divided by the video length of the corresponding step. If the Repetition Index is 1.5, the user watched the whole step once, and half of it once more.
- **Backjump Frequency**: (*Number of backward jumps*). Users jump backward on the video to watch the part that is demonstrated quickly or unclearly. We count the number of backward jumps that occurred while watching a step.
- **Pause Frequency**: (*Number of pauses*). Users pause the video to transfer the content in the tutorial to their application if it needs much attention. If there are frequent pauses, it may indicate that the step is hard to digest and to be transferred to their context at once. We count the number of pauses that occurred in a step. We do not consider the duration of pauses as it highly overlaps with the Execution Time Index.
- **Miss Rate**: *(Proportion of users who missed a step at first but followed it later)*. If a step is not clearly shown in the video, sometimes users skip the step at first. We define the Miss Rate as the proportion of users who missed a step at first but followed it later. A high Miss Rate indicates that users can easily miss the step.

Measure	Definition (*video time*: a duration of a step in video)
Execution Time Index	Time taken to follow a step / *video time*
Repetition Time Index	Total time of a step being watched / *video time*
Backjump Frequency	Number of backward jumps
Pause Frequency	Number of pauses
Miss Rate	The proportion of users who missed a step at first but followed it later
Re-follow Rate	The proportion of users who re-followed a step after proceeded to the next steps

Table 4: Definition of six measures that portray the difficulty of each step.

Measure	Definition
Relevant Steps	Previous steps that users followed after watching the current step to complete the step
Referring Rate	The proportion of users who followed previous steps again to proceed with the current step
Continued Rate	The proportion of users who only watched the current step to proceed with the step (i.e., 1 - Referred Rate)

Table 5: Definition of three measures related to relevancy of each step.

- **Re-follow Rate**: (*Proportion of users who re-followed a step after proceeded to the next steps*). If a step was not completed in the users' context, they might revisit and perform the action again even after they moved on to the later steps. We define the Re-follow Rate as the proportion of users who revisited the step and performed it again. A high Re-follow Rate means many users go back to the step and follow it again, indicating a high chance where the step could not be properly done.

4.1.2 Step relevancy. We defined three measures about relevancy of each step: Relevant Steps, Referring Rate, and Continued Rate. Relevant step information can help learners who get stuck in a certain step, by suggesting they check other related steps again. To help learners decide whether they should check the relevant steps, we also define Referring Rate and Continued Rate. Below we describe each measure in detail (Table 5).

- **Relevant Steps**: When users get stuck in a certain step, they sometimes try previous steps again to help them complete the step. We define the previous steps that are followed after watching the current step to complete the current step as Relevant Steps. Figure 4 shows an example scenario describing Relevant Steps.
- **Referring Rate**: Referring Rate means the proportion of users who followed the previous steps after watching the current step, to complete the current step. In other words, it is the proportion of users who produced the Relevant Steps. It indicates how relevant the Relevant Steps are.
- **Continued Rate**: In contrast to the Referring Rate, the Continued Rate means the proportion of users who only watched the current step to proceed with the step. In other words, it is (1 - Referring Rate).

4.2 Methodology

We describe the methodology we used to compute the above measures for each step from the collected interaction data.

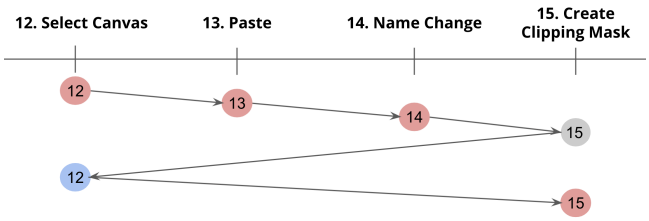


Figure 4: An example scenario where the relevant step of step 15 is step 12. After a user followed the step 12, 13, and 14, he is now on step 15. However, the user was not able to complete it. The user jumped back to step 12 and then followed it again. Then, he came back to step 15 and followed the step. (red: followed, gray: watched but not followed, blue: followed again).

4.2.1 Removing actions that are unrelated to the task. After collecting interaction data—video interaction logs (play, pause, jump backward/forward) in synchronization with the software usage logs—we first processed the software usage logs to remove the actions that are unrelated to tasks. The History Log feature in Photoshop extracts actions done on Photoshop including actions that are not directly related to the main tasks, such as auto-saving files or quitting the application. Thus, we removed log entries that are not related to the tasks.

4.2.2 Identifying the followed and skipped steps. To compute the Execution Time Index, Miss Rate, Re-follow Rate, and Relevant Steps, we need to identify when and which steps were followed or skipped. For example, we need to know when a user successfully followed a step to compute the Execution Time Index.

To identify if a user followed or skipped a step, we first define *baseline actions* as actions done in tutorial videos and *baseline timestamps* as the starting timestamps in the tutorial video of the corresponding baseline action (Figure 5). To get the baseline actions, we followed the tutorials exactly the same on our Photoshop, checking which action is being logged in the History Log feature.

For the baseline timestamp, we manually recorded the timestamp where each action began to be described in the video by watching the tutorial videos.

After setting up the baseline actions and baseline timestamps, we developed an algorithm that detects whether a user followed or skipped a step from the interaction logs (Algorithm 2). The algorithm detects that a user **followed** a step 1) if they performed a baseline action after passing, 2) but still nearby the corresponding baseline timestamp; threshold values in Algorithm 1 determine the range of "nearby". The algorithm detects a user **skipped** a step if they did not follow the step but followed the next step. The algorithm detects a user **added** an action if the action does not exist in the video or it exists but is not considered as *followed*.

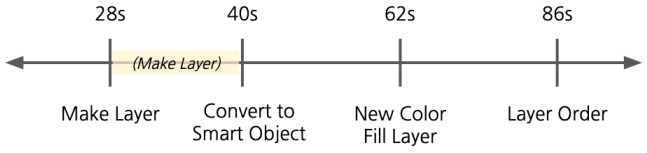


Figure 5: An example of baseline actions and their corresponding baseline timestamps. A baseline action named `Make Layer` is explained in the tutorial video from 28s to 40s, so the baseline timestamp of `Make Layer` is 28s.

Algorithm 1: IsFollowed

Input: A list of baseline timestamps, $T = t_0, ..., t_n$
A list of baseline actions, $A = a_0, ..., a_n$
A current video timestamp, t
An action performed by a user, a
An index of the expecting action that needs to be done, i
An index of the most recent action that a user has watched, w
Output: **True** if the action is a followed action, **False** otherwise

$thresholdPrevious, thresholdAfter \leftarrow$ Thresholds of video timestamp offsets
if $i \leq w$ **then**
  $thresholdPrevious \leftarrow 20$;
  $thresholdAfter \leftarrow 20$
else
  $thresholdPrevious \leftarrow 5$;
  $thresholdAfter \leftarrow 15$;
if $a = a_i$ **then**
  if *(($i \leq w$* ***and*** *a_i is unique in A)* ***or***
    ($t \geq t_i - thresholdPrevious$ ***and***
    ($t \leq t_i + thresholdAfter$ ***or*** *$i = n$)) then*
    return True
return False

Algorithm 2: Action State Detection

Input: A list of baseline timestamps, $T = t_0, ..., t_n$
A list of baseline actions, $A = a_0, ..., a_n$
A current video timestamp, t
An action performed by a user, a
An index of the expecting action that needs to be done, i
An index of the most recent action that a user has watched, w
An index of the previous followed action, p
A list of user logs, $L = [(state_0, action_0), ..., (state_m, action_m)]$; `/* state is either 'followed', 'added', or 'skipped' */`
Output: $L = [(state_0, action_0), ..., (state_{m+1}, action_{m+1})]$

if *a is not in A* **then**
  $L \leftarrow$ L + ($`added', p$);
  return
; `/* Check if a user followed the expecting action or previous actions */`
$j \leftarrow i$
while $j > 0$ **do**
  if *isFollowed(T, A, t, a, i, w)* **then**
    $L \leftarrow L +$ ($`followed', j$);
    return
  $j \leftarrow j - 1$
; `/* Check if a user skipped an action and followed a further action */`
$j \leftarrow i + 1$
while $j < len(L)$ **do**
  if *isFollowed(T, A, t, a, i, w)* **then**
    for $k = i$ *to* j **do**
      $L \leftarrow L +$ ($`skipped', k$)
    $L \leftarrow L +$ ($`followed', j$);
    return
  $j \leftarrow j + 1$
$L \leftarrow L +$ ($`added', p$);

4.2.3 Computing the measures for each step. Among the six measures regarding the difficulty of steps, we computed the Execution Time Index, Repetition Time Index, Backjump Frequency, and Pause Frequency for each user per step. Then, we averaged the values among users per step and regarded the averaged value as a representative value of each step. We computed Miss Rate, Re-follow Rate, and the three measures of step relevancy (i.e., Relevant Steps, Referring Rate, and Continued Rate) per step.

To estimate the difficulty of each step, for each of the six measures, we identified the steps with a value higher than the third quartile (i.e., 75%) of all steps. For example, we identified a step with high Execution Time Index by comparing its value to the third quartile of the Execution Time Index values of all steps. We apply the quartile method since it is widely used to classify data into subgroups considering the distribution [5].

Additionally, for the measures that could be computed per user (i.e., Execution Time Index, Repetition Time Index, Backjump Frequency, and Pause Frequency), we computed the third quartile of each measure within a step among users in the same group (i.e., Novice or Experienced). This is to set multiple thresholds to identify if a user is having difficulty. For example, if a novice user's Execution Time Index of a step is exceeding the third quartile of novice users in the same step, we could assume that the user is undergoing difficulty in the step.

4.3 Results

Through the analysis, we computed 1) the six difficulty-related measures for each step and for each user per step, and 2) the three step relevancy-related measures for each step. Table 6 shows the average values of the six difficulty-related measures across the step for each tutorial. Except for the Miss Rate, the difference between novice and expertise group was statistically significant (Mann-Whitney Test, $p < 0.01$ or $p < 0.05$), showing the reliability of the measures used (Table 6). It indicates that novice users showed more behavior of having difficulties than the experienced users.

We describe several examples of the results below. Table 7 shows example measures of steps that exceed the third quartile of all steps, which might indicate that the step is likely to be more difficult. Table 8 shows the third quartile values of each measure for each step, which serve as threshold values when detecting users' confusing moments. Table 9 shows examples of Relevant Steps, Referring Rate, and Continued Rate. We can see that even though steps `Move` and `Select Canvas` from the Logo tutorial all have at least three Relevant Steps, their significance could be different as the Referring Rates differ substantially (41% vs. 8%).

From the analysis, we could also see that Miss Rate demonstrated steps that have certain properties that make them easy to miss. For example, 39% of participants missed the `Drag Selection` on the Logo tutorial, which was passing fast and not noticeable. Re-follow Rate captured steps that need attention. For example, 60% of users followed `Layer Order` again in the Geometry tutorial. Positioning the layers in the right order was important but many participants did it incorrectly at first.

5 SOFTVIDEO

We present SoftVideo, a prototype system that provides step information, gives feedback to learners on their progress, and provides help to overcome confusing moments (Figure 1). SoftVideo provides step information such as the name of an action, and the duration and estimated difficulty of each step in the timeline (Figure 1(a)). It gives feedback to users about their progress by letting them know if they completed or missed a step (Figure 1(b)) and detecting when they struggle (Figure 1(c)). Finally, it presents help suggestions such as to slow down the pace, replay the step, or see relevant steps when they struggle (Figure 1(d)-(f)).

There are three components in SoftVideo that are powered by the analyzed data (Section 4): Estimated difficulty of each step, criteria for detecting users' confusing moments, and relevant steps that are suggested when they struggle. All the information is determined based on the group the user belongs to (i.e., Novice or Experienced) so that the system provides customized help. User can enter their level of experience before they start watching the video (Figure 1(g)).

Measure	Expertise	Logo	Geometry	Avg.
Execution Time Index	Novice	5.4*	6.3*	5.9
	Experienced	4.0*	5.5*	4.7
Repetition Time Index	Novice	1.82*	1.8*	1.81
	Experienced	1.43*	1.53*	1.48
Backjump Frequency	Novice	1.79**	1.24**	1.52
	Experienced	0.76**	1.10**	0.93
Pause Frequency	Novice	1.60*	1.07*	1.34
	Experienced	1.27*	0.58*	0.93
Miss Rate (%)	Novice	5.1%	4.5%	4.8%
	Experienced	3.2%	5.7%	4.5%
Re-follow Rate (%)	Novice	16.8%*	15.4%	16.1%
	Experienced	8.3%*	13.0%	10.7%

Table 6: Mean values of the six difficulty-related measures among all steps. In general, novice users show more behavior of having difficulties than experienced users. For each measure, the table shows if the difference between the novice and experienced groups was statistically significant (*: p<.05, **: p<.01, Mann-Whitney Test) for each measure.

Tutorial	Step	Measures
Logo	1. Polygonal Lasso	Execution Time Index, Repetition Time Index, Pause Frequency, Backjump Frequency, Re-follow Rate
	7. Drag Selection	Repetition Time Index, Miss Rate
Geometry	6. Ellipse Tool	Repetition Time Index, Pause Frequency, Backjump Frequency
	37. Move	Miss Rate, Re-follow Rate

Table 7: Examples of difficulty-related measures that exceed the third quartile of all steps. The more measures there are, the higher the probability that the step will be difficult.

Step	Expertise	Execution Time Index		Pause Frequency	
		Mean	*3rd Quartile*	*Mean*	*3rd Quartile*
1. Polygonal Lasso *(Logo)*	Novice	5.07	6.75	4.71	6
	Experienced	4.8	5.86	3.25	4.25
6. Ellipse Tool *(Geometry)*	Novice	6.05	7.46	3.6	5
	Experienced	5.14	6.76	2.58	2.25

Table 8: Examples of mean and the third quartile (threshold) values for the Execution Time Index and Pause Frequency. The threshold values are used to detect when users are going through confusing moments.

Tutorial	Step	Relevant Steps	Referring Rate	Continued Rate
Logo	8. Move	4. Color Range, 1. Polygonal Lasso, 7. Drag Selection	41%	59%
	12. Select Canvas	8. Move, 4. Color Range, 7. Drag Selection	8%	92%
Geometry	13. Blending Change	12. Move, 6. Ellipse Tool, 7. Select Canvas	27%	73%
	25. Set Shape Layer Stroke	–	0%	100%

Table 9: Examples of Relevant Steps, Referring Rate and Continued Rate. Up to the top three relevant steps are shown in order.

Icons	Meanings	Measures
	Users spent **more time** in this step compared to other steps.	Execution Time Index
	Users watched this step **repeatedly more** than other steps.	Repetition Time Index
	Users did **backward jumps frequently** at this step more than other steps.	Backjump Frequency
	Users **paused frequently** at this step more than other steps	Pause Frequency
	There are relatively many users who **missed** the step.	Miss Rate
	There are relatively many users who **followed again** the step.	Re-follow Rate

Table 10: Icons that depict the difficulty of each step, and their corresponding meanings and measures.

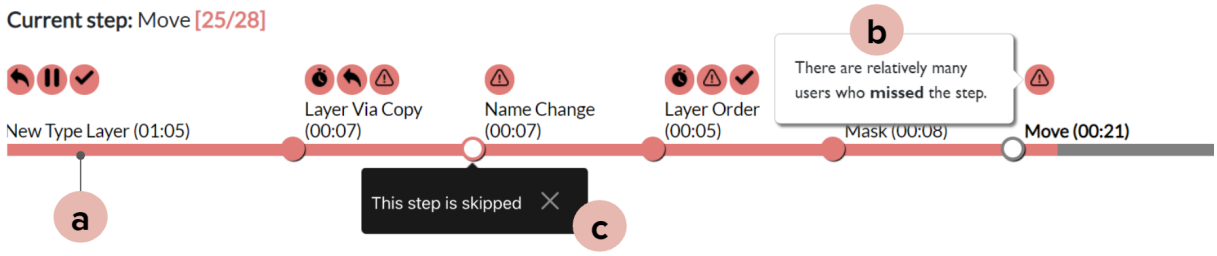


Figure 6: (a) The Timeline shows step information; action name, its duration, and difficulty. Each icon represents a message related to the difficulty (Table 10). In the `Layer Via Copy`, users spent more time compared to other steps, did many backward jumps, and there are relatively many users who missed the step at first. (b) Users can hover on an icon to see its meaning. (c) The timeline also gives real-time feedback on users' progress. The user skipped the step `Name Change` and thereby is warned.

5.1 Step Information

SoftVideo provides a timeline that shows step information in the tutorial video (Figure 6). The timeline is segmented into steps and each step is shown with the Photoshop action name and its duration, which is reflected in its length in the timeline. The timeline display of step descriptions has been introduced by other systems (e.g., [28]), but we additionally provide characteristics of each step that represent the difficulty of a step. With the six difficulty-related measures (Section 4.1.1), SoftVideo presents icons for the measures with values that exceed the third quartile of all steps. Table 10 shows the icons and their meanings, and corresponding measures. For example, if a step is shown with the pause icon, it means that users paused frequently at the step more than other steps. Thus, users can estimate the difficulty or complexity of a step by skimming through the icons shown in the timeline. We chose to present such potentially useful indicators rather than a single quantified difficulty level, so that users can have control over how they leverage the given information.

5.2 Real-time Feedback

SoftVideo gives real-time feedback to users on their progress by tracking both the video and the application logs. First, it lets users know if they completed a step or not with our action detection algorithm, described in Section 4.2.2. If a user follows a step in their application correctly, then the circle of the step gets filled. If a user misses a step and proceeds to the next step, the circle remains unfilled and the user is warned (Figure 6(c)). Second, it detects when a user is facing difficulties. If any of the six measures exceeds its threshold value (Section 4.2.3), the system alerts users by asking "*Are you stuck?*" and presents appropriate help suggestions, which are described in the next section (Figure 7-right).

5.3 Help Suggestions

When SoftVideo detects users undergoing confusing moments, it suggests users to 1) slow down the pace, 2) replay the step, or 3) go back to relevant steps. Users can slow down the video pace to x0.5 or x0.75 by clicking the button (Figure 1(d)), or replay the step by clicking the circle on the timeline (Figure 1(e)). SoftVideo also

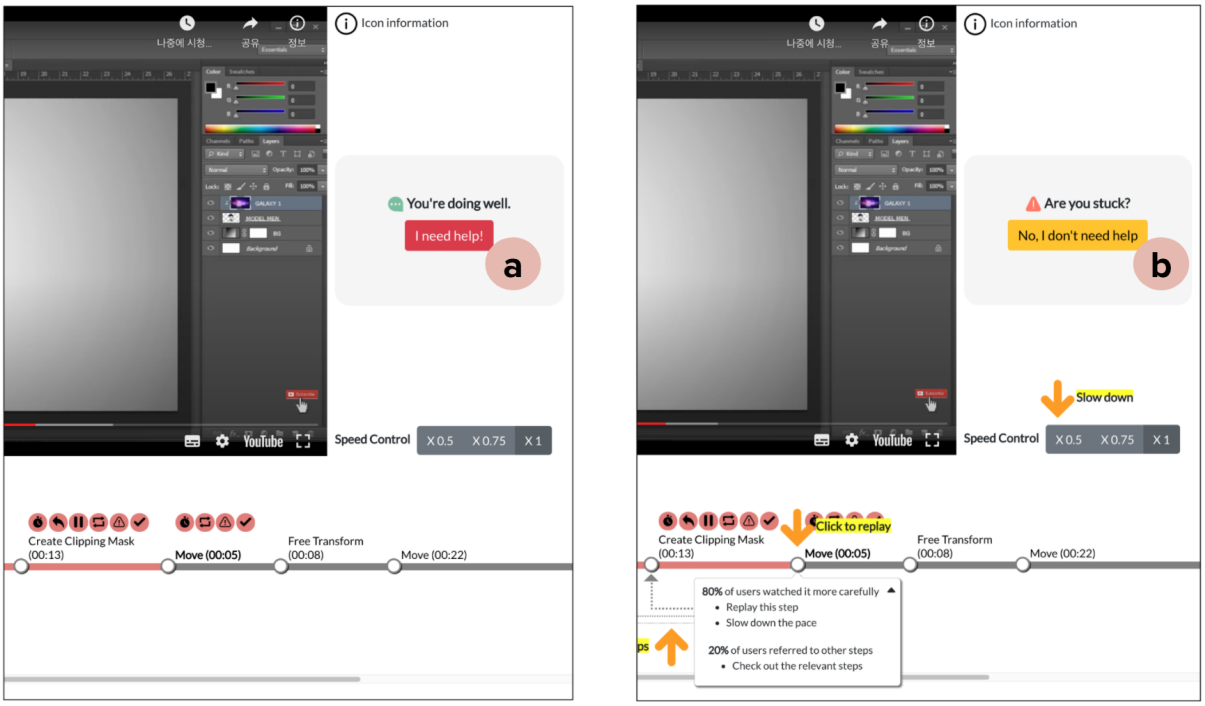


Figure 7: (Left) A user is following the tutorial video. Once the system detects that the user may be confused or struggling, (Right) SoftVideo presents action suggestions as help. Users can also proactively (a) request to see the help (b) or close the help.

suggests users to check relevant steps (Figure 1(f)). The arrow to a relevant step is thicker if more users followed the step after watching the current step. To help users better decide if they should check the relevant steps or not, SoftVideo presents the ratio of users who only watched the current step to complete it and users who watched and followed previous steps to complete it (Section 4.1.2). This is to help users with decision making rather than giving pressure to check relevant steps. If a user moves to other steps, the suggested help gets closed.

Users can also request to see help by clicking the "*I need help!*" button (Figure 7(a)) or close the help suggestions by clicking the "*No, I don't need help*" button (Figure 7(b)). This is to make sure users access necessary help suggestions on demand (or dismiss unnecessary information) in case the algorithm failed to detect their confusing moments.

5.4 Implementation

We implemented SoftVideo using React.js, HTML, and CSS for the front-end web interface, and Node.js and Firebase for the back-end server. The implementation mostly follows the system used in the data collection study (Section 3.1). It additionally runs the action detection algorithm (Algorithm 2) in real-time to track users' progress and runs the data analysis pipeline (Section 4) in real-time for computing the Execution Time Index, Repetition Time Index, Backjump Frequency, and Pause Frequency measures to detect users' confusing moments.

6 USER EVALUATION

We evaluated the feasibility of using data-driven information and the effectiveness of SoftVideo through a study. Specifically, the goals of our evaluation were (1) to see how participants think about and use the step information when performing tasks, and (2) to assess the effect of real-time feedback and help suggestions on improving the user experience of software tutorial videos.

6.1 Participants

We recruited 30 (22 male, 8 female, mean age 23.8) participants from an academic institution through an online community posting, including 23 novice and 7 experienced users for Photoshop. The level of Photoshop expertise were determined in the same manner as in Section 3.2. People who participated in the data collection study were excluded from this recruitment. Each participant was assigned to one of the two tutorial videos used in the data collection study. We assigned the participants equally for each tutorial; 15 (11 novice, 4 experienced) were assigned to the Logo tutorial while the remaining 15 (12 novice, 3 experienced) were assigned to the Geometry tutorial. Participants were compensated 20,000 KRW (approximately USD 17) for their participation in a 80-minute-long study.

6.2 Study Procedure

The study took place face-to-face, following the COVID-19 guidelines: participants had to wear masks and sanitize their hands before

using computers. Windows and doors were open and an air conditioner was turned on to keep the room ventilated. We sanitized the utilities after each session.

Participants were first asked to complete a pre-task survey about their experiences in using Photoshop and how they interpret each of the six message types (Table 10) to make sure they become familiar with the messages. We then introduced a Photoshop tutorial video to participants and asked them to choose images to be used based on their preference. After explaining how to use SoftVideo, one researcher set up their expertise level (novice or experienced) in SoftVideo based on the pre-task survey result and entered the path to the Photoshop History Log file for real-time tracking. Participants were then asked to follow the given tutorial video using SoftVideo. Once participants completed the main task, we conducted a survey about their experience and a semi-structured interview to get more detailed feedback. Each participant was provided with two monitors; one for the tutorial video (SoftVideo) and the other for Photoshop.

We chose not to do a comparative study as SoftVideo is a complex system with multiple novel features: a comparative study cannot clearly uncover the source of differences observed, and it is unclear what a convincing baseline might be. Rather, we focus on observing and analyzing how participants use SoftVideo in a realistic task. We logged the number and the timestamp of detected confusing moments, help requests and help dismissals made by participants, and their usage of help suggestions.

7 RESULTS

Below we summarize the main findings and usefulness of SoftVideo with respect to each feature.

7.1 Step Information

Participants were able to estimate the difficulty of steps with the number of icons shown in the timeline. In general, they felt that the number of icons implied the difficulty of a step (perceived accuracy = 3.73/5, std=0.98). Being able to know about the difficulty of steps affected them in a few different ways, which we report below.

7.1.1 Participants planned their behavior and level of concentration according to the difficulty of steps. Participants were able to plan their action and level of concentration by looking at the difficulty of upcoming steps. They planned their pauses on the video depending on the difficulty (P3, P4, P22, P26). P22 said, *"I put my fingers on the space bar in advance when facing difficult steps so that I can be ready to pause."* Similarly, P3 said, *"when there were no icons, I tried to watch the step at once until the end without pauses."* Participants not only planned their pauses but also controlled the speed of the video playback (P1, P10, P14, P23). P1 said, *"I was able to prepare myself for upcoming steps by slowing down the pace whenever I saw many icons."* Even if they did not perform an explicit action to be prepared, they adjusted their level of concentration based on the difficulty (P11, P13, P17, P19, P23, P24, P27, P29). P13 said, *"When there were no icons, I was relaxed and watched the step in a relaxing way. However, when there were many icons, I focused more."*

Participants' experiences in early steps affected their planning strategy. P14 said, *"I found myself being able to watch and follow at the same time when there were two or fewer icons. After experiencing that I pause a lot during steps with three or more icons, I started to slow down the pace of the video right before such steps came up."* P6 built their own understanding of the icons through the earlier steps which made them perform certain actions prior to watching steps with particular icons. P6 said, *"I learned that there was a pause icon whenever the step required me to enter in some parameters like width and height. After experiencing it, I was able to know when similar actions (i.e., setting values) are coming (when I saw the pause icon) and so I was able to perform them in advance."*

7.1.2 Step-wise difficulty information increased the level of safety and gave hints when they struggle. When participants faced confusing moments, they checked to see icons and felt relieved to see many icons on the step (P4, P5, P8, P12, P16, P18, P19, P27). P27 said, *"I felt relieved to see many icons when I was struggling because I knew it was not only me and the problem is the step itself."* It also happened when participants came back to a certain step after having done it differently or missed it. P18 said, *"I didn't notice the icons at first, but when I revisited a step to do it again, I could see many icons and was able to know that there were many similar users like me."*

The difficulty level also gave hints on how to overcome confusing moments—whether they should look into the step in more detail or watch other steps. P7 said, *"When I struggled, I watched the step more carefully if there were many icons. In contrast, if there were few icons, I realized something went wrong in previous steps, not the current step, so I watched previous steps."*

7.1.3 Differences in the perceived usefulness between the messages. Although most participants perceived the icon count as an indicator of step difficulty, there were differences in perceived usefulness between the messages. Participants rated the usefulness of messages as follows (ordered by score): Pause Frequency (4.03/5), Repeat Index (3.73/5), Revisited Rate (3.73/5), Execution Time (3.63/5), Backjump Frequency (3.6/5), and Missed Rate (2.7/5). Pause Frequency might have been the most useful because knowing how to split a step is important in following tutorial videos. P27 said, *"I tended to pause if there was the pause icon when I wasn't sure about when to pause."* On the other hand, Missed Rate might have been the least useful because participants might have felt that there are small chances of missing a step, partially due to SoftVideo's feature of letting users know if they have missed a step. In general, participants said it was helpful to see step information (3.7/5, std=1.3).

7.2 Real-time Feedback

We report how participants felt about real-time feedback on their progress and automatic detection of confusing moments.

7.2.1 Letting users know about their progress. With the feedback SoftVideo provides, participants were able to identify missed steps (P4, P6, P9, P18) as well as steps that they performed differently from the tutorial video (P7, P12, P14). P9 said, *"I noticed a difference between the image on the tutorial and the image on my application. Then I noticed that there was a step that I missed due to an alert SoftVideo gave. I was able to go back to the missed step and follow it."* SoftVideo also let users know about steps that they thought they followed but not actually because they behaved differently. P7 said, *"I thought I followed the step* `Move` *but it didn't appear to be so, so I checked it again. I realized that I didn't press 'Ctrl' while doing the*

action." Participants mentioned that the real-time feedback on the progress encouraged them to follow the tutorial more meticulously (P11) and it made following along more enjoyable as it felt like solving a series of quests (P25). Participants said the feature was helpful in general (3.67/5, std=1.3).

7.2.2 Detecting confusion moments. Overall, participants felt that SoftVideo detected their confusing moments accurately. On a scale of 1 to 5, with 1 being early and 5 being late about the timing of SoftVideo's confusion detection, participants rated 2.9 (better if closer to 3, std=0.92). P6 said, *"I thought it detected quite well. I was struggling at a step of doing 'Ctrl+T' and the system detected it right away."* On average, SoftVideo detected 17.83 confusing moments per user (min: 1, max: 29). Participants closed 2.76% of the suggested help and requested to see help 0.77 times additionally on average. For about 32% out of 543 detection and requested cases, participants utilized at least one of the suggested help, which we discuss next.

7.3 Help Suggestions

We report the usage of help suggestions by SoftVideo and how participants found information of relevant steps helpful.

7.3.1 How participants used suggested help. Among the three help suggestions SoftVideo provides (i.e., speed control, repeating a step, and relevant steps), participants repeated a step most frequently (114), followed by checking the suggested relevant steps (46) and slowing down the pace (13). Participants might have repeated a step a lot because it is what most users are familiar with, checking if they have missed anything and figuring out why it does not work on their application by watching over and over. On the other hand, they rarely slowed down the pace when faced with difficulties. P25 said, *"I didn't use the speed control because the part that needs attention only lasted a few seconds. I didn't want it to be slower for the entire step."*

7.3.2 How seeing relevant steps was helpful. Participants reported that seeing the suggested relevant steps was helpful in overcoming confusing moments (P2, P5, P7, P9, P11, P14, P16, P19, P23). It helped them by suggesting steps that they should watch again. P2 said, *"When I knew I made a small mistake, I jumped back to 5 seconds before by using the left arrow key on the keyboard. However, when I wasn't sure what caused a problem, seeing the relevant steps was helpful."* In particular, if one of the relevant steps was pointing to a step that they have missed, they perceived it as an important step and went back to the step to follow it (P5, P7, P14, P19). It not only helped participants follow the step they have missed, but also to re-follow the step that they have followed before. P11 said, *"I was able to catch up right away after watching a relevant step. Even though I followed the step, there was something I pressed in a wrong way."*

Some participants perceived the relevant steps as "similar steps", and transferred the knowledge of the step to the current step. P8 mentioned *"I was able to relate the information from a relevant step. I remembered how I completed the step, so I thought I could do this step in a similar way."* Another interesting usage was that it helped participants acquire the knowledge of the software, by looking at which steps are frequently related. P25 said, *"It helped me a lot in understanding how to use Photoshop in general. I was able to know which actions are related and which should be done for other actions to be done."* Also, with relevant steps participants reported feeling safe because even if they failed to follow a step, there are alternatives that they could try (P26, P27).

However, unlike our expectations, the Referring Rate and Continued Rate were rarely used. Nearly all participants mentioned that they did not look at the numbers. P7 said, *"I didn't see the numbers at all. If there was at least one relevant step, I checked it out no matter how many referred to it."* Although the Referring Rate and Continued Rate were not used in deciding whether they should watch relevant steps, some participants used the information to adjust their concentration level on the relevant steps (P2, P24). P24 said, *"If the Referred rate was about 80%, I watched it normally. If it was higher than 85%, I paid more attention. If it was 92% or higher I paid extra attention and watched it carefully."*

7.4 Other Feedback

Participants also appreciated the basic timeline that shows the name and duration of each action. It helped them learn about the sub-goal of each step (P4, P8, P17) and made it easier to navigate the video (P1, P9, P13, P29). Seeing the action name was helpful because participants were able to expect which menu they should click (P29), especially when the same step appears again later (P19). Overall, participants found SoftVideo helpful in following along the tutorial content (4.17/5, std=0.87). Moreover, they preferred using SoftVideo compared to the basic video-only interface (*'I'd prefer to use this system to the basic video-only interface.'* (5-point Likert scale): 4.13/5, std=0.97).

8 DISCUSSION, LIMITATIONS, AND FUTURE WORK

In this paper, we investigated the feasibility of enhancing software tutorial videos with data-driven information. In this section, we discuss considerations, limitations, and possible future work of using collective interaction data.

8.1 Utilizing Synchronized Interaction Data of Both Software and Tutorial Video

Synchronized interaction data of how a user uses both the software and the tutorial video possess much more potential than just two single data sources. It allows for more accurate inference of the user's current state and more personalized support. For example, our Execution Time, Missed Rate, and Revisited Rate measures are induced from (and are only made possible by) synchronized data of both the software and the tutorial. Using such metrics extracted from synchronized data, in addition to metrics obtained from video interaction logs (i.e., Repetition Index, Backjump Frequency, and Pause Frequency) which have been shown to be relevant with video difficulty [36], we were able to detect whether the user is experiencing difficulty in following the tutorial. Similarly, previous work also showed that utilizing additional logs such as physiological data collected from smartwatches can significantly improve the video difficulty detection [10]. Likewise, if we only utilized one data source or if the data was not synchronized, the impact of SoftVideo could have been less significant.

8.2 Users' Trust and Interpretations on Data-driven Information

SoftVideo's data-driven information shows the collective behavior of a number of users who have worked toward a shared goal. How users perceive the meaning of information might be different from user to user. Participants from our study built up their trust towards the system and came up with their own understanding of how to interpret the provided information as they used the system. P26 said, *"I found out that those steps do not have icons because I could easily follow the video while watching at the same time."* Similarly, P4 said, *"It was cool that I actually paused a lot in steps with many icons."* This shows that their trust towards the system grew as they used the system and their experience aligned with the presented information. After understanding how the presented information matches their context, participants built their own techniques to interpret and follow subsequent steps (e.g., to pause the video at steps with three or more icons). It also shows that giving users control to selectively leverage useful signals rather than presenting a single answer predicted by the system allowed them to build trust and make their own interpretations.

8.3 Availability of Interaction Data and Its Privacy Implications

In order to utilize synchronized interaction data of both software and tutorial video, it is essential to first consider how to obtain software interaction data. For example, our work uses Photoshop as an instance of software, which enables tracking software usage logs through its History Log feature. Modern software applications such as AutoCAD [3] or Fusion360 [1] also provide history logs so that users can track their progress and easily revert to a particular action. For software with no history logs or API for them, accessibility APIs [18, 40] or computer vision techniques [4, 37, 39, 46] could be used to reverse-engineer the software interactions. Augmenting open-source software such as GIMP [23] could be another possible solution.

When capturing interaction data, privacy issues should be carefully considered. Unlike videos that are published publicly on online platforms, the software is often where users work privately. Previous work suggests that when users acknowledge that there are enough benefits provided, users' perceived privacy concerns may be alleviated [29, 44], but still sensitive personal information or assets (e.g., file names) can be recorded in the software usage logs. Potential solutions include automatically filtering out such information or giving users control by allowing them to review and filter what gets shared.

8.4 Leveraging Richer Interaction Data

In our work, we collected pause, play, and jump as video interaction data and Photoshop action names as software interaction data. Future work could look into leveraging richer interaction data. For example, playback speed change or volume control of videos might capture important or non-important parts of the video. Also, users' Undo and Redo behavior on the software can be used [15, 42], as it may imply important moments of the video such as confusing parts or the parts where people explore. With such data, it may be possible to identify steps that are optional or steps where users can branch out and be more creative about. As such, more extensive interaction data could improve the accuracy in revealing important points in tutorial videos. Moreover, analyzing the interaction data with respect to users' expertise level or quality of outcome can enable tailored support according to expertise level or goals.

8.5 More Support for Learners and Authors of Educational Videos

SoftVideo demonstrates how utilizing interaction data can enhance the learning experience of software tutorial videos. Extending this idea, future systems can provide further support to learners. As people use the system, the system can give adaptive information to users. The system can control the amount and the content of the information in a personalized way by identifying what information a user needs. For example, a certain part of the video can be only shown to users who encounter a certain type of difficulties. Also, although we set the third quartile as a universal metric when defining the difficulty of a step or detecting users' confusion, future work can investigate adaptive techniques for identifying the user's state and providing more personalized experiences.

Furthermore, our system could be beneficial for authors of educational videos. For example, an author of an instructional video can identify where users struggle a lot or which steps users miss frequently so that they can improve the video or provide additional explanations. Visual analytics tools of how users learn through instructional videos might give insights into understanding users and improving the content as well.

With our public dataset of synchronized interaction logs of the tutorial videos and the software, we expect that it could facilitate a further understanding of how users learn from software tutorial videos. We expect that it will enable future research in data-driven video-based learning.

9 CONCLUSION

This paper presents SoftVideo, a data-driven interface for improving the learning experience of software tutorial videos. SoftVideo helps users plan ahead before watching a step, gives feedback on their progress, and presents help suggestions when they struggle. We analyzed collective interaction logs of a tutorial video in synchronization with the software to provide the difficulty of each step, detect users' confusing moments, and suggest relevant steps. A user study showed that data-driven information allowed participants to plan their behavior of following the tutorial, feel relieved, and overcome confusing moments. We believe that leveraging richer interaction data could further enrich the learning experience of both instructional videos and complex software.

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2020R1C1C1007587) and Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2021-0-01347, Video Interaction Technologies Using Object-Oriented Video Modeling).

REFERENCES

[1] Fusion 360. 2022 (accessed February 9, 2022). *Autodesk*. https://www.autodesk.com/products/fusion-360/overview

[2] YouTube Data API. 2022 (accessed February 9, 2022). *YouTube*. https://developers.google.com/youtube/v3/docs/videos

[3] AutoCAD. 2022 (accessed February 9, 2022). *Autodesk*. https://www.autodesk.com/products/autocad/overview

[4] Nikola Banovic, Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2012. Waken: Reverse Engineering Usage Information and Interface Structure from Software Videos. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 83–92. https://doi.org/10.1145/2380116.2380129

[5] Christopher Brinton, Swapna Buccapatnam, Mung Chiang, and H. Vincent Poor. 2016. Mining MOOC Clickstreams: On the Relationship Between Learner Video-Watching Behavior and In-Video Quiz Performance. *IEEE Transactions on Signal Processing* 64, 1–1. https://doi.org/10.1109/TSP.2016.2546228

[6] Peter Brusilovsky. 1998. Methods and techniques of adaptive hypermedia. In *Adaptive hypertext and hypermedia*. Springer, 1–43.

[7] Minsuk Chang, Ben Lafreniere, Juho Kim, George Fitzmaurice, and Tovi Grossman. 2020. Workflow Graphs: A Computational Model of Collective Task Strategies for 3D Design Software. In *Graphics Interface*.

[8] Hsiang-Ting Chen, Li-Yi Wei, Björn Hartmann, and Maneesh Agrawala. 2016. Data-Driven Adaptive History for Image Editing *(I3D '16)*. Association for Computing Machinery, New York, NY, USA, 103–111. https://doi.org/10.1145/2856400.2856417

[9] Pei-Yu Chi, Sally Ahn, Amanda Ren, Mira Dontcheva, Wilmot Li, and Björn Hartmann. 2012. MixT: Automatic Generation of Step-by-Step Mixed Media Tutorials. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 93–102. https://doi.org/10.1145/2380116.2380130

[10] Jinhan Choi, Jeongyun Han, Woochang Hyun, Hyunchul Lim, Sun Young Huh, SoHyun Park, and Bongwon Suh. 2019. Leveraging Smartwatches to Estimate Students' Perceived Difficulty and Interest in Online Video Lectures. In *Proceedings of the 2019 11th International Conference on Education Technology and Computers*. 171–175.

[11] Konstantinos Chorianopoulos. 2013. Collective intelligence within web video. *Human-centric Computing and Information Sciences* 3, 1 (2013), 1–16.

[12] Firebase Realtime Database. 2022 (accessed February 9, 2022). *Firebase*. https://firebase.google.com

[13] Jonathan D. Denning, William B. Kerr, and Fabio Pellacini. 2011. MeshFlow: Interactive Visualization of Mesh Construction Sequences. Vol. 30. Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/2010324.1964961

[14] Himel Dev and Zhicheng Liu. 2017. Identifying Frequent User Tasks from Application Logs. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces* (Limassol, Cyprus) *(IUI '17)*. Association for Computing Machinery, New York, NY, USA, 263–273. https://doi.org/10.1145/3025171.3025184

[15] W. Keith Edwards, Takeo Igarashi, Anthony LaMarca, and Elizabeth D. Mynatt. 2000. A Temporal Model for Multi-Level Undo and Redo. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology* (San Diego, California, USA) *(UIST '00)*. Association for Computing Machinery, New York, NY, USA, 31–40. https://doi.org/10.1145/354401.354409

[16] Leah Findlater, Karyn Moffatt, Joanna McGrenere, and Jessica Dawson. 2009. Ephemeral Adaptation: The Use of Gradual Onset to Improve Menu Selection Performance *(CHI '09)*. Association for Computing Machinery, New York, NY, USA, 1655–1664. https://doi.org/10.1145/1518701.1518956

[17] C. Ailie Fraser, Joy O. Kim, Hijung Valentina Shin, Joel Brandt, and Mira Dontcheva. 2020. Temporal Segmentation of Creative Live Streams. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376437

[18] C. Ailie Fraser, Tricia J. Ngoon, Mira Dontcheva, and Scott Klemmer. 2019. RePlay: Contextually Presenting Learning Videos Across Software Applications *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300527

[19] Floraine Grabler, Maneesh Agrawala, Wilmot Li, Mira Dontcheva, and Takeo Igarashi. 2009. Generating Photo Manipulation Tutorials by Demonstration. In *ACM SIGGRAPH 2009 Papers* (New Orleans, Louisiana) *(SIGGRAPH '09)*. Association for Computing Machinery, New York, NY, USA, Article 66, 9 pages. https://doi.org/10.1145/1576246.1531372

[20] Tovi Grossman, George Fitzmaurice, and Ramtin Attar. 2009. A Survey of Software Learnability: Metrics, Methodologies and Guidelines. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. Association for Computing Machinery, New York, NY, USA, 649–658. https://doi.org/10.1145/1518701.1518803

[21] 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

[22] Version history of Adobe Photoshop. 2022 (accessed February 9, 2022). *Adobe*. https://helpx.adobe.com/photoshop/using/whats-new.html

[23] GNU image manipulation program. 2022 (accessed February 9, 2022). *GIMP*. https://www.gimp.org

[24] Kimia Kiani, Parmit K. Chilana, Andrea Bunt, Tovi Grossman, and George Fitzmaurice. 2020. "I Would Just Ask Someone": Learning Feature-Rich Design Software in the Modern Workplace. In *2020 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. 1–10. https://doi.org/10.1109/VL/HCC50065.2020.9127288

[25] Kimia Kiani, George Cui, Andrea Bunt, Joanna McGrenere, and Parmit K. Chilana. 2019. Beyond "One-Size-Fits-All": Understanding the Diversity in How Software Newcomers Discover and Make Use of Help Resources. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3290605.3300570

[26] 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

[27] 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

[28] Juho Kim, Phu Tran Nguyen, Sarah Weir, Philip J. Guo, Robert C. Miller, and Krzysztof Z. Gajos. 2014. Crowdsourcing Step-by-Step Information Extraction to Enhance Existing How-to Videos. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 4017–4026. https://doi.org/10.1145/2556288.2556986

[29] Seoyoung Kim, Arti Thakur, and Juho Kim. 2020. *Understanding Users' Perception Towards Automated Personality Detection with Group-Specific Behavioral Data*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376250

[30] René F. Kizilcec, Chris Piech, and Emily Schneider. 2013. Deconstructing Disengagement: Analyzing Learner Subpopulations in Massive Open Online Courses. In *Proceedings of the Third International Conference on Learning Analytics and Knowledge* (Leuven, Belgium) *(LAK '13)*. Association for Computing Machinery, New York, NY, USA, 170–179. https://doi.org/10.1145/2460296.2460330

[31] Geza Kovacs. 2016. Effects of In-Video Quizzes on MOOC Lecture Viewing. In *Proceedings of the Third (2016) ACM Conference on Learning @ Scale* (Edinburgh, Scotland, UK) *(L@S '16)*. Association for Computing Machinery, New York, NY, USA, 31–40. https://doi.org/10.1145/2876034.2876041

[32] Benjamin Lafreniere, Tovi Grossman, and George Fitzmaurice. 2013. Community Enhanced Tutorials: Improving Tutorials with Multiple Demonstrations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 1779–1788. https://doi.org/10.1145/2470654.2466235

[33] Ben Lafreniere, Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2014. Investigating the Feasibility of Extracting Tool Demonstrations from In-Situ Video Content. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 4007–4016. https://doi.org/10.1145/2556288.2557142

[34] Guo Li, Tun Lu, Jiang Yang, Xiaomu Zhou, Xianghua Ding, and Ning Gu. 2015. Intelligently Creating Contextual Tutorials for GUI Applications. In *2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom)*. IEEE, 187–196.

[35] Nan Li, Lukasz Kidzinski, Patrick Jermann, and Pierre Dillenbourg. 2015. How Do In-video Interactions Reflect Perceived Video Difficulty? *Proceedings of the European MOOCs Stakeholder Summit 2015* (2015), 112–121. http://infoscience.epfl.ch/record/207968

[36] Nan Li, Łukasz Kidziński, Patrick Jermann, and Pierre Dillenbourg. 2015. MOOC Video Interaction Patterns: What Do They Tell Us? In *Design for Teaching and Learning in a Networked World*. Springer International Publishing, 197–210.

[37] Lingfeng Bao, Jing Li, Z. Xing, Xinyu Wang, and Bo Zhou. 2015. Reverse engineering time-series interaction data from screen-captured videos. In *2015 IEEE 22nd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*. 399–408.

[38] Zipeng Liu, Zhicheng Liu, and Tamara Munzner. 2020. Data-Driven Multi-Level Segmentation of Image Editing Logs. 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, 1–12. https://doi.org/10.1145/3313831.3376152

[39] Justin Matejka, Tovi Grossman, and George Fitzmaurice. 2011. Ambient Help. 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, 2751–2760. https://doi.org/10.1145/1978942.1979349

[40] Justin Matejka, Tovi Grossman, and George Fitzmaurice. 2013. Patina: Dynamic Heatmaps for Visualizing Application Usage. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 3227–3236. https://doi.org/10.1145/2470654.2466442

[41] Aadhavan M. Nambhi, Bhanu Prakash Reddy, Aarsh Prakash Agarwal, Gaurav Verma, Harvineet Singh, and Iftikhar Ahamath Burhanuddin. 2019. Stuck? No Worries! Task-Aware Command Recommendation and Proactive Help for Analysts. In *Proceedings of the 27th ACM Conference on User Modeling, Adaptation and Personalization* (Larnaca, Cyprus) *(UMAP '19)*. Association for Computing Machinery, New York, NY, USA, 271–275. https://doi.org/10.1145/3320435.3320477

[42] Mathieu Nancel and Andy Cockburn. 2014. Causality: A Conceptual Model of Interaction History. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 1777–1786. https://doi.org/10.1145/2556288.2556990

[43] Cuong Nguyen and Feng Liu. 2015. Making Software Tutorial Video Responsive *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 1565–1568. https://doi.org/10.1145/2702123.2702209

[44] Chanda Phelan, Cliff Lampe, and Paul Resnick. 2016. It's Creepy, But It Doesn't Bother Me. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 5240–5251. https://doi.org/10.1145/2858036.2858381

[45] Photoshop. 2022 (accessed February 9, 2022). *Adobe*. https://www.adobe.com/products/photoshop

[46] 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 *(UIST '11)*. Association for Computing Machinery, New York, NY, USA, 135–144. https://doi.org/10.1145/2047196.2047213

[47] Luca Ponzanelli, Gabriele Bavota, Andrea Mocci, Rocco Oliveto, Massimiliano Di Penta, Sonia Haiduc, Barbara Russo, and Michele Lanza. 2019. Automatic Identification and Classification of Software Development Video Tutorial Fragments. *IEEE Transactions on Software Engineering* 45, 5 (2019), 464–488. https://doi.org/10.1109/TSE.2017.2779479

[48] Tanmay Sinha, Patrick Jermann, Nan Li, and Pierre Dillenbourg. 2014. Your Click Decides Your Fate: Inferring Information Processing and Attrition Behavior from MOOC Video Clickstream Interactions. *EMNLP Workshop on Modelling Large Scale Social Interaction in Massive Open Online Courses*. https://doi.org/10.3115/v1/W14-4102

[49] Xu Wang, Benjamin Lafreniere, and Tovi Grossman. 2018. Leveraging Community-Generated Videos and Command Logs to Classify and Recommend Software Workflows. 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, 1–13. https://doi.org/10.1145/3173574.3173859

[50] Min web browser. 2022 (accessed February 9, 2022). *Min browser*. https://minbrowser.org/

[51] Whale web browser. 2022 (accessed February 9, 2022). *Naver*. https://whale.naver.com/

[52] Kuldeep Yadav, Ankit Gandhi, Arijit Biswas, Kundan Shrivastava, Saurabh Srivastava, and Om Deshmukh. 2016. ViZig: Anchor Points Based Non-Linear Navigation and Summarization in Educational Videos. In *Proceedings of the 21st International Conference on Intelligent User Interfaces* (Sonoma, California, USA) *(IUI '16)*. Association for Computing Machinery, New York, NY, USA, 407–418. https://doi.org/10.1145/2856767.2856788

[53] Saelyne Yang and Juho Kim. 2020. What Makes It Hard for Users to Follow Software Tutorial Videos?. In *Proceedings of HCI Korea 2020*. The HCI Society of KOREA, South Korea, 531–536.

[54] Tom Yeh, Tsung-Hsiang Chang, Bo Xie, Greg Walsh, Ivan Watkins, Krist Wongsuphasawat, Man Huang, Larry S. Davis, and Benjamin B. Bederson. 2011. Creating Contextual Help for GUIs Using Screenshots. 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, 145–154. https://doi.org/10.1145/2047196.2047214

[55] Tingshao Zhu, Russ Greiner, G Haubl, Bob Price, and Kevin Jewell. 2005. Behavior-based Recommender Systems for Web Content. In *Proceedings of the Wokshop Beyond Personalization 2005, in conjunction with the International Conference on Intelligent User Interfaces IUI'05*. Citeseer, 83–88.