Automatic Instructional Video Creation from a Markdown-Formatted Tutorial

Peggy Chi
Nathan Frey
Google Research
howtocut@google.com

Katrina Panovich
Google
howtocut@google.com

Irfan Essa
Google Research, Georgia Institute of Technology
howtocut@google.com

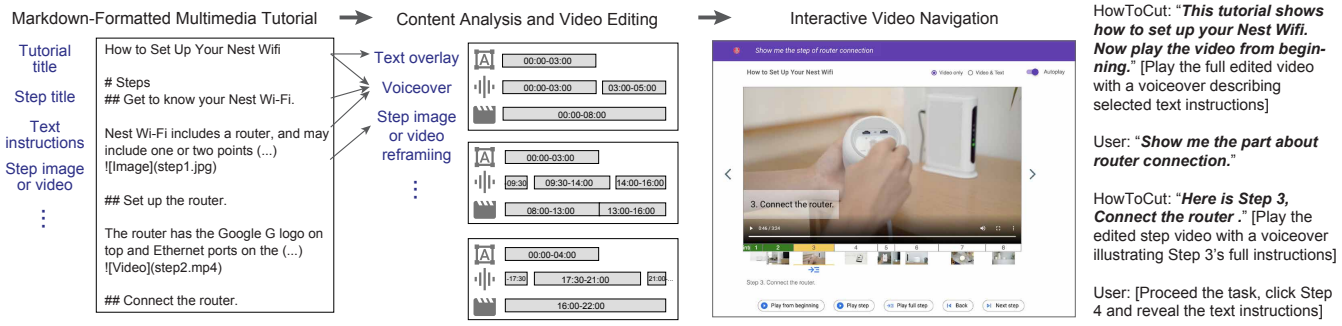

Figure 1: Given a Markdown-formatted document of a step-by-step tutorial, HowToCut automatically generates a video that presents the instructions both verbally and visually. HowToCut selects and enhances the text instructions to a synthesized voiceover and text overlays. It makes automatic editing decisions on the timing and camera movements to align the step images or videos with the voiceover. Viewers can follow the tutorials as an interactive video via our conversational UI, which shows different levels of information. *Tutorial source: Google Nest, "How to set up your Nest Wifi."*

ABSTRACT

We introduce HowToCut, an automatic approach that converts a Markdown-formatted tutorial into an interactive video that presents the visual instructions with a synthesized voiceover for narration. HowToCut extracts instructional content from a multimedia document that describes a step-by-step procedure. Our method selects and converts text instructions to a voiceover. It makes automatic editing decisions to align the narration with edited visual assets, including step images, videos, and text overlays. We derive our video editing strategies from an analysis of 125 web tutorials and apply Computer Vision techniques to the assets. To enable viewers to interactively navigate the tutorial, HowToCut's conversational UI presents instructions in multiple formats upon user commands. We evaluated our automatically-generated video tutorials through user studies (N=20) and validated the video quality via an online survey (N=93). The evaluation shows that our method was able to effectively create informative and useful instructional videos from a web tutorial document for both reviewing and following.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
UIST '21, October 10–14, 2021, Virtual Event, USA
© 2021 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-8635-7/21/10.
https://doi.org/10.1145/3472749.3474778

CCS CONCEPTS

• **Human-centered computing → Human computer interaction (HCI)**.

KEYWORDS

Video creation; instructional videos; how-to videos; web document; voiceover; creativity tools.

ACM Reference Format:
Peggy Chi, Nathan Frey, Katrina Panovich, and Irfan Essa. 2021. Automatic Instructional Video Creation from a Markdown-Formatted Tutorial. In *The 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21), October 10–14, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3472749.3474778

1 INTRODUCTION

Tutorials illustrate how to complete specific tasks through a step-by-step procedure. Topics of a How-To tutorial can range from cooking, repair, sports, Do-It-Yourself (DIY), to many other categories [44, 50]. The demand for online tutorials has vastly increased [45, 49, 50], especially during the recent COVID-19 pandemic with an increase of skill learning and in-home projects [41]. Thanks to the easy access to recording devices and online platforms, tutorial creators have built long-time community efforts to share and co-edit procedural knowledge [31, 45]. As of year 2020, wikiHow offers over 232,000 web articles in 19 languages by 1,000 experts globally. Their content covers a wide range of themes, illustrated by 4 million images or supporting files [53]. iFixit offers 75,000 manuals for consumer electronics and home appliances [20].

Instructables has collected more than 325,000 tutorials in 15 years on DIY topics [26].

It is common to present tutorials as either a text-driven document with step images or videos, or a stand-alone video that describes the task procedure [45, 50]. Prior studies have shown that tutorial formats benefit viewers differently in learning: Tutorial documents are efficient for glancing through a procedure; videos are effective in showing the exact actions for task following, especially when presented as steps [9, 48, 59]. With a voiceover, videos further support viewers who prefer listening to the verbal instructions while observing the actions [50]. Existing digital platforms make it easy to share instructions as a web document or a video, however, it is less common that tutorial authors share both formats due to the editing efforts required. This forces learners to choose one or the other medium to follow a task.

In this paper, we introduce HowToCut, an automatic approach for converting step-by-step text and visual instructions in a document to a video that can be interactively reviewed (see Figure 1). We focus on online tutorials composed in the Markdown format, a popular markup language that enables tutorial authors to collaboratively edit multimedia instructions [17, 20, 53], derived from the revolution of Wikipedia and open source communities. HowToCut analyzes a Markdown-formatted tutorial and makes automatic video editing decisions. It partitions and enhances text instructions, generates a synthesized voiceover, and processes and presents step images and videos to reveal the step-by-step structure. Tutorial followers can review the tutorial as an interactive video in our conversational UI. They can navigate the narrated video based on steps and explore the detailed instructions shown in different formats when following a task.

We evaluated our automatically-generated video results from 40 existing web tutorials. To gather audience feedback, we conducted two user studies with a total of 20 participants and an online survey with 93 respondents. The findings suggested that our method was able to effectively create informative videos from a tutorial for viewers to review and follow instructions. Our work makes the following contributions:

- An automatic approach to generate instructional videos from a Markdown-formatted tutorial and provide interactive navigation.
- Methods to automatically convert a structural tutorial document to a video, which enhances text instructions for a voiceover and presents visual instructions with editing effects, including text overlays, looping, zooming, and camera motion.
- An evaluation of automatically generated videos from web tutorials with DIY enthusiasts and general audience.

2 RELATED WORK

HowToCut builds on prior work of computational techniques that convert documents to new presentations, generate videos, and create interactive tutorials. We review and discuss our relationship with the related efforts in these topics.

2.1 Document Content Conversion

To enable wider audience to consume web content, there is a considerable amount of research in converting a document to a multimedia format, including speech, slideshows, and videos. Research in accessibility demonstrates methods to generate verbal description of a web page [2, 18] or a graphical user interface [58] based on its document or UI hierarchy. By prioritizing high-level information, the narration better guides users to follow a document. To visualize content, recent work automatically segments Wikipedia or web articles and animates as a slideshow [7, 60] or a non-narrated video [8, 29]. When a text-driven document does not include adequate visual materials, keywords from paragraphs are used to find relevant stock images to enhance a slideshow [33, 56]. The advancement of natural language understanding also introduces new ways for document navigation, such as questioning and answering [25, 43].

These prior arts extract document content, often designed for linear navigation of non tutorials. We focus on presenting *instructional content* for viewers who aim to follow procedural instructions. Our approach considers the amount of information in a tutorial given its document structure, while enabling viewers to navigate instructions and select presentation formats interactively.

2.2 Automatic Video Creation

Creating effective videos is an effort-consuming process. Post editing can be especially challenging, as it requires iterations to organize footage, place cuts, and apply visual effects. Prior research has proposed a variety of methods to automate video editing fully or partially [4, 19, 47], often tailored for specific domains. Berthouzoz et al. provided an automatic tool for placing cuts and transitions on interview footage [3]. Without footage, video frames can be automatically synthesized from text [33, 42, 51], documents [8, 29], or physical activities [22, 55]. For physical tasks such as cooking and DIY projects, video footage can cover a wide range of topics, making it difficult for automatic techniques to accurately edit. By involving human authors to provide high-level annotations, semi-automatic approaches can speed up the editing process [10, 46]. Video data can further support human editing, such as using transcripts of conversational videos [11, 32] or the audio signals [39].

Our work automates the video creation process for a specific domain: procedural knowledge. Instead of making automatic edits on raw footage and dialogues, we focus on instructional articles that contain text instructions and visual materials covering various levels of details. Our method selects and edits content based on the document structure for guiding viewers' attention, which differs from prior work that generates non-interactive videos.

2.3 Tutorial Creation and Consumption

It requires intensive time and efforts to create a tutorial, especially to edit and present concise guidance [45]. To reduce the editing difficulty, researchers have proposed computational techniques to present procedural knowledge as diagrams [1], documents [15], how-to videos [10, 46], or mixed-media tutorials [9, 38]. These require source footage and automatic or human-provided annotations that existing online tutorials do not provide. Tools can also convert public tutorials to new presentations, including a mixed-media interface for skimming [48], a voice-based interface for physical tasks [5, 6], or augmented experiences [36, 37, 57]. We share the goal of making instructional content useful based on tutorial followers' preferences and needs. Our method maintains the tutorial

Figure 2: For a tutorial from the same author, we compared the presentation of its web article (left) and the instructional video (right). For an article, we reviewed the step information (index, title, detailed text instructions, and image or video). For a video, we reviewed the text overlays, video cuts, and transcript if available. *Tutorial source: wikiHow, "How to Make Apple Juice" and its YouTube video, shared with permission from wikiHow.*

structure and provides the content breakdown for interactive navigation. While we do not claim novelty in the tutorial presentation, we present a unique automatic technique of converting a tutorial document to a useful format.

3 EXISTING PRACTICES OF INSTRUCTIONAL VIDEOS

To understand the common editing practices of instructional videos, we examine 125 publicly-available web tutorials and their corresponding videos. We derive a set of design principles for converting a step-by-step tutorial to a How-To video, and confirm our observation with video producers.

3.1 Video Analysis

Recent studies identified common editing techniques applied in instructional videos, including text overlays, visual annotations or zooming to highlight details, and verbal cues in the voiceover [10, 48, 50]. As we are interested in the correlations between a tutorial document and an instructional video, we collected examples from YouTube channels of popular instructional platforms that offer user guides of physical or software tasks, including (in alphabetical order) Allrecipes.com, Google Help, iFixit, Instructables, and wikiHow. We looked for video tutorials where their video description explicitly included the supplementary document and confirmed if their instructions were aligned (see Figure 2). We asked six in-house paid raters to annotate the time codes of each step in a source tutorial as shown in its corresponding video.

Table 1 shows the results. On average, each rater spent 15 minutes to map a 15-step tutorial to a 97.5-second video. Given the video segments mapped to the tutorial steps, we observed how the *voiceover* and the *visuals* align. We found that most videos are designed to be concise. Each step may include a few instructional sentences that describe a specific action, followed by brief details or tips. The same step in the web article often contains more detailed notes that are excluded in the video. Videos often include the same structure and step titles from the documents, but are often shorter in length.

For videos that have a voiceover, the narration typically runs continuously without a long break or silence. A step commonly starts with a key word, such as the step number ("*Step one*") or the ordering ("*First*", "*Next*", and "*Finally*"). Most of the voiceover exactly describes what is shown in the video. For example, a step "*Open the app*" would show the exact action tapping the app on a phone screen. For another example, when hearing "*Add the essential oil*" in the voiceover, the video shows the exact motion dropping oil into a bowl. We observed similar editing techniques suggested by prior work [10, 50], including text or graphical annotations that highlight critical information.

3.2 Discussion with Video Producers

To understand the common practices from professionals, we conducted a one-hour interview with two professional video producers who created a series of instructional videos for hardware devices in our organization. We learned about their existing practices and verified our findings. To produce a video, professionals start from collecting instructional details, writing a script, and planning detailed visuals that match the script. Once the script is finalized, they capture a set of footage of the step-by-step procedure with the

Table 1: We collected 125 web tutorials of 12 categories from popular sources and observed video presentation techniques.

	# of tutorials	# of steps in document			Ave. video duration	Example title
		AVE	MIN	MAX		
Car	3	13.33	5	19	173.00	How to Disconnect a Car Battery
Cooking	35	15.26	4	35	102.50	How to Melt Cheddar Cheese
Crafting	16	14.06	7	24	116.10	How to Do String Art
Entertainment	5	12.60	7	18	99.80	How to Build a Tower of Cards
Fashion	12	15.58	9	24	79.70	How to Make a Quick Dutch Braid
Gardening	4	14.75	7	22	84.80	How to Cut Fresh Rosemary
Health	21	14.86	8	20	72.30	How to Measure Hand Size
Home	15	18.50	11	29	103.00	How to Kill Ants Outside
Repair	3	18.00	11	30	112.30	How to Fix Thigh Holes in Jeans
Sports	7	14.29	8	30	99.60	How to Shoot a Basketball
Software	2	4.00	4	4	88.00	Google Meet in Gmail quick start
Pet	2	12.50	8	17	79.50	How to Bathe Your Puppy
Complete Set	**125**	**15.14**	**4**	**35**	**97.50**	

duration of each step in mind. In the post-production phase, professionals never alter the speed of the voiceover in order to provide a consistent tone. They carefully align the visuals with the voiceover illustrating specific instructions. Each video has a consistent visual style, so that the audience can easily identify the brand and details. Such a production process is time- and budget-consuming, often involving multiple iterations.

3.3 Design Guidelines

Based on the findings from our video analysis and prior work, we propose four design guidelines for tools that convert a tutorial document to an instructional video.

- *Focus on instructions.* A tool should maintain the original instructions without incorrectly manipulating the content. It should reveal the step-by-step structure in the same ordering as shown in the source document.
- *Align voiceover and visuals.* Verbal instructions should match the visuals in a video. Use text overlays to provide additional anchors and highlight critical information.
- *Provide concise content.* Different from a web tutorial that commonly includes detailed information and tips, a video should concisely describe the instructions with an adequate amount of details to engage viewers.
- *Consistent editing.* Consistent editing styles make content easy to follow, including the same speech rate, text annotations, pacing, and visual effects. When necessary, apply editing techniques such as zooming and panning to guide viewers' attention.

4 TASK FOLLOWING WITH HOWTOCUT

We present HowToCut, an automatic approach that converts instructional content from a Markdown-formatted tutorial to an interactive How-To video. We focus on tutorials that describe a step-by-step procedure, where each step contains both text instructions and step images or videos. We develop an end-to-end solution that extracts the multimedia assets and the hierarchical information from a Markdown document, and automatically makes both temporal and visual decisions of placing content in a video. Our generated videos present the extracted step images and videos with a synthesized voiceover of the text instructions. Tutorial followers can review the instructions via HowToCut's conversational user interface, which provides a step-based video timeline and navigation options (see Figure 3).

To help us illustrate the functionality, assume that a user, Maurie, is looking for a tutorial to set up her Wi-Fi router at home. Maurie identifies a web tutorial with a title that matches her interests and decides to follow the instructions. She opens HowToCut's UI on a smart display and specifies the web article. She then sees a three-minute video in the UI (see Figure 3a), composed of eight steps (see Figure 3b). Here, the video is automatically generated by HowToCut, which retrieves the source document via the tutorial site's API to fetch the instructions and re-format the content as a video that can be reviewed interactively.

HowToCut presents a tutorial overview as a video by visually illustrating the steps with a synthesized voiceover that narrates the key text instructions. Via the action list on the UI (see Figure 3e-2), Maurie plays the video from beginning, which starts with an introduction ("*To set up your Nest Wifi*"), followed by the steps in a linear sequence with verbal connections in between (including "*First*", "*Next*", and "*Finally.*") Each step shows a title (e.g., "*Connect the router*" for Step 3 in Figure 3a) with more information in the voiceover. Maurie could focus on the actions shown in the video frames, while listening to the voiceover to follow the details.

After seeing the video, Maurie has her device and tools ready for installation. She wants to know more instructions of the step before starting the work, so she activates HowToCut with a hotword and asks it to "*Show the step of router connection*" (see Figure 3e-1). The UI jumps to Step 3 and talks back with the step title, "*Here is Step 3, Connect the router*". It then plays the full step video that HowToCut automatically created, which narrates the full text instructions and zooms to focus on the actions (see Figure 3d). Later, Maurie wants to confirm the details of password settings. She identifies Step 4 from the animated thumbnail. Instead of playing the video, she reveals the text with the side-by-side view (see Figure 3c) and quickly finds the information. With this UI, Maurie has the full control to review instructions of a specific step in the format based on her needs when following a task.

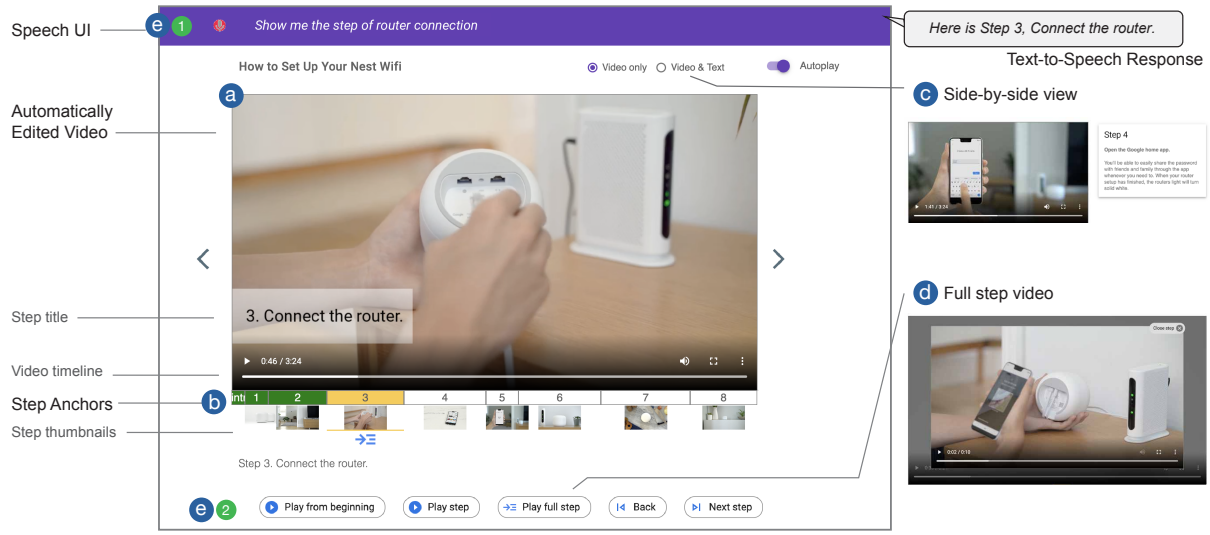

Figure 3: HowToCut's user interface presents the automatically-generated video of a web tutorial (a) and its step overview (b), including the step numbers and animated thumbnails. Viewers can choose to reveal the text instructions (c) and more details of a step as a longer video (d). Our UI traces the user review progress and includes a conversational agent, which receives user commands through voice input (e-1) or GUI (e-2) and responses with a synthesized speech. *Tutorial source: Google Nest, "How to set up your Nest Wifi."*

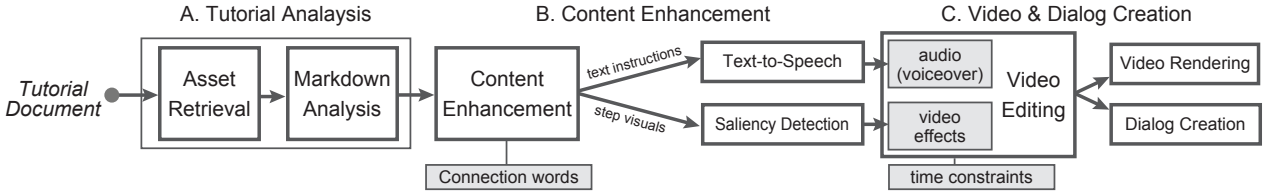

Figure 4: HowToCut's video creation pipeline: Given a Markdown-formatted tutorial, HowToCut retrieves and analyzes the instructional content (A). It converts text instructions into a sequence of Text-to-Speech audio files and performs saliency detection on the step images and videos (B). Given the output duration constraints, it makes automatic edits and creates a main video, full step videos, and a dialog script for interactive navigation (C).

5 VIDEO CREATION FROM A MARKDOWN-FORMATTED TUTORIAL

HowToCut automatically makes video edits in both the temporal and visual domains to present instructions from a Markdown-formatted tutorial (see Figure 4). First, it extracts the instructions and hierarchy. For text instructions, HowToCut partitions text instructions and adds narration for connection; for visual instructions, it performs saliency detection for reframing. With the synthesized voiceover from text and the step images or videos, it organizes assets to create an instructional video and a dialog script for navigation. Below we describe our video creation algorithm.

5.1 Tutorial Analysis

HowToCut takes an input of a Markdown-formatted text document that annotates instructional content in the Markdown language [17] (see Figure 5a). Using a Markdown renderer, HowToCut retrieves the document as a tree structure for parsing. Popular tutorial platforms–such as wikiHow [54] and iFixit [21]–commonly define their customized Markdown formats and provide public guidelines for community contributors to follow. For examples, the heading markup `##` denotes a step title, and `![Video](step2.mp4)` presents a video file given a filename. Based on these public guidelines, we convert the structural content to the step-by-step tutorial scheme we defined (see Figure 5b). We assume that a tutorial document D is composed of a linear list of K sections, denoted as $D = \{sec_k, k \in K\}$. A section sec_k can be an introduction or a tutorial step, which includes:

- An ordered list of M *text instructions*, each as a sentence, denoted as $T_k = \{text_i, i \in M\}$ where $M > 0$. Based on the Markdown format, $text_i$ can be annotated as a step title, a tip, a list item or other element.
- An ordered list of N *visual instructions* such as step images and videos, denoted as $V_k = \{visual_j, j \in N\}$ where $N > 0$. Each item

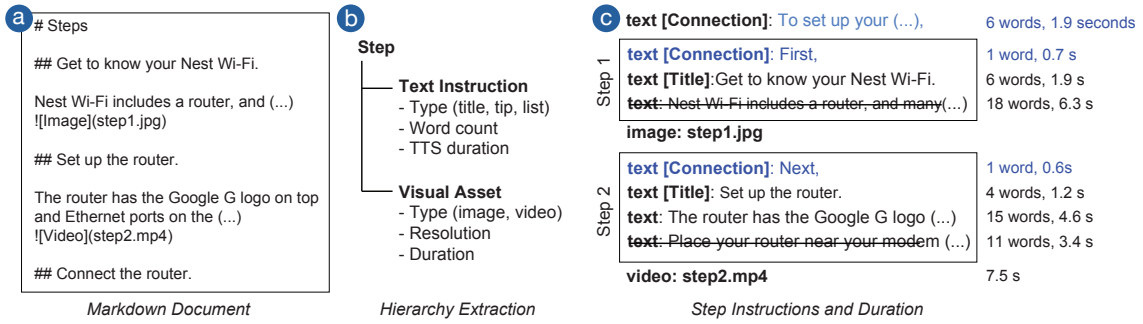

Figure 5: Given a Markdown tutorial document (a) and the step annotation scheme we defined (b), HowToCut segments the step-by-step content and partitions text instructions, each of which is synthesized as a Text-to-Speech audio file of a fixed duration (c). Based on the duration of a step image or video, HowToCut selects the amount of text instructions to be included in the voiceover.

has a duration d_j^{visual}. For a video file, the duration is its video length. We assign a minimum duration (3 seconds) to an image.

Using this scheme, our pipeline parses each Markdown node to build a tutorial structure and partition the text into sentences, as Figure 5c shows.

5.2 Content Enhancement

HowToCut enhances a tutorial's structure to help viewers follow the instructions. Language studies suggest that *transitional words* – such as "first, then, finally"– build connections between ideas, especially to emphasize event sequences [52]. However, tutorial guidelines may avoid using transitional words in document writing to provide concise instructions [54]. The connection is shown via the document format, including step indices and blocks (see Figure 2 left). To convert written text instructions into a voiceover with verbal connections, we define two bags of transitional words, including (1) a tutorial goal used in the introduction section (e.g., "*Follow this tutorial to learn how to (...)*" from the document title) and (2) step indices or progress (e.g., "*Step one*", "*Step two*", ... or "*First*", "*Next*", "*Then*", and "*Finally*"). Based on the tutorial structure, we expand the text instructions of each sec_k as $T'_k = \{text_i, i \in M'\}$, where $M' \geq M$. For example, the introduction (sec_0) calls out the tutorial title, where the first step (sec_1) starts with "*First*" and the following step (sec_2) starts with "*Next*". HowToCut generates a Text-to-Speech (TTS) voiceover for each sentence $text_i \in T'_k$, each has a voiceover duration d_i^{text}.

5.3 Video Creation

HowToCut places the step-by-step text and visual instructions into a series of video scenes. Inspired by existing How-To videos that we learned from our formative analysis, we aim to concisely present the visual assets with a continuous voiceover that describes the instructions. For each step, we select critical content and apply video editing techniques to compose the materials.

5.3.1 Content Selection. HowToCut selects a subset of text instructions as the voiceover and visual instructions as the graphical frames from each step. Learned from our analysis, a written tutorial commonly contains detailed text instructions and tips that might not be necessarily ideal for narration. To make an instructional video concise, we design a strategy to include all key information (step indices and titles), while adding extra supportive content based on the duration of visual materials. To formalize our selection algorithm, for each section $sec_k \in D$:

(1) Select the critical text instructions as the voiceover, $VO_k = \forall text_i[transition, title] \in T'_k$. Acquire its total duration as $d_k^{VO} = \sum_{i=0}^{S} d_i^{text} + d^{pause} \times S$.
(2) Retrieve the duration of visual instructions V_k in this step as $d_k^{visual} = \sum_{j=0}^{N} d_j^{visual}$.
(3) If $d_k^{VO} < d_k^{visual}$ (i.e., the voiceover is shorter than the visuals that would cause an audio gap), add the next text instruction $text_{S+1} \in T'_k$ with an updated duration $d_k'^{VO} = d_k^{VO} + d_{S+1}^{text}$. Repeat this until $d_k'^{VO} \geq d_k^{visual}$ or $\forall text_i \in T'_k$ have been added to VO_k.
(4) If $VO_k \subsetneq T'_k$ (not all text instructions are included in the voiceover), prepare a voiceover of the full step video $VO_k^{full} = \forall text_i \in T'_k$ that narrates the full text instructions. Acquire its duration $d_k^{full-VO}$.

We found it unnecessary to perform summarization or partial sentence selection using language models to avoid alternating the instructions. We aim to provide a consistent mechanism for users to navigate the content, although models trained on specific topics could potentially enhance our constraint-based method.

5.3.2 Visual Composition. Once collecting a series of instructions, HowToCut organizes the assets onto the video timeline and makes editing decisions, including video effects and text overlays (see Figure 6a). The audio track of a video contains the voiceover of all steps $\forall VO_k \in D$. As for the video frames, HowToCut composes timestamped visual layers: For each step, the base layer is a step image or a video. It adds a text layer for the title or list item using predefined graphical layouts. For example, a title is shown at a consistent location for viewers to quickly access the information. A text layer is visible while its corresponding voiceover is playing.

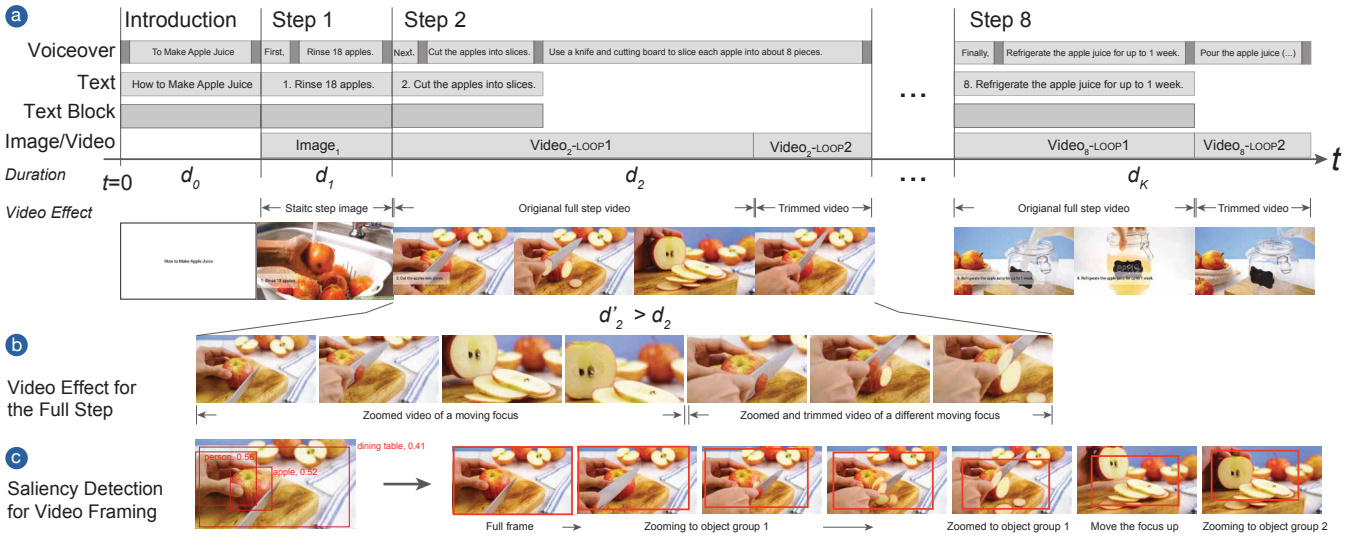

Figure 6: HowToCut makes automatic editing decisions to place the voiceover on the timeline and align with the visual assets of a step-by-step procedure from a document (a). It applied editing techniques to present the concise instructions, including looping, text overlays, and zooming (b). Using Computer Vision techniques, we detect salient objects and adjust the camera motion to reframe the video (c). *Tutorial source: wikiHow, "How to Make Apple Juice," shared with permission from wikiHow.*

5.3.3 Video Editing and Reframing. By default, we present the step images or videos without adding visual effects until the voiceover of the step is complete ($d_k'^{VO}$). If their duration falls short of the voiceover ($d_k^{visual} \leq d_k'^{VO}$), we extend the image duration or loop the video until $d_k'^{VO}$. However, these basic effects might not be optimal, especially for the instructions with a long voiceover. A static image without transitions can easily lose viewers' attention; video looping can distract or confuse viewers. Prior work has suggested the effectiveness of video transitions for instructions, such as zooming to an author-labeled object [10]. Therefore, we post-edit a full step video with zooming effects to engage viewers in the instructions (see Figure 6b).

For each step image and video, we perform face and object detection to identify salient objects or focal points [13, 14]. We sample the video every 0.1 seconds to annotate the frame. Each label has an identity name (e.g., face, apple, bowl), a bounding box (the position of the object shown in the frame), and a confidence score (ranged between the minimal threshold until high confidence as 1). Similar to prior work that automatically reframes an edited video based on scene content [12, 30, 34], our pipeline ranks the objects in a frame given their confidence scores, region sizes, and locations.

We render the static image or loop the step video into a video v_k at length $d_k^{full-VO}$ of this step. We then dynamically move the camera view of v_k to focus on salient objects from the labeling results (see Figure 6c). We design a camera motion path to highlight the region of interest (ROI) of salient objects that have a major occupancy toward the center of a scene: The camera view starts from the full frame at $t = 0$, smoothly zooms to one salient object, moves on to another one or two objects, and zooms back to the full frame at the end of the video. The intention is similar to how a person moves the camera to multiple objects sequentially to illustrate the scene. In this way, a result video v_k' can have a more dynamic visual focus with a voiceover that describes detailed instructions. Our post-editing method considers the source video resolution and constrains to zoom to at most 50% of the frame. The camera path is smoothed based on a constrained velocity to avoid abrupt or fast motion. This approach supports multi-shot or looped source videos by moving to multiple ROIs continuously across different shots. If no ROI is identified, the video is looped without reframing.

5.4 Dialog Creation and Interactive Navigation

HowToCut provides additional metadata for progress tracking and instruction navigation. For each step, it annotates the start and end time in the edited video, along with the full text instructions that can be revealed in the UI (see Figure 3c). Our UI provides a conversational agent with predefined dialogue templates (e.g., "*Want me to replay the video?*" and "*Here is Step [index], [title]*") and navigation commands (e.g., "*Play the video*" and "*Go to Step 4*") similar to recent work [48]. In addition, we track users' video playback progress and dynamically adjust the navigation menu, such as "*Play from beginning*" and "*Play this step*".

Finally, we provide a basic Questioning and Answering (Q&A) mechanism using Semantic Reactor, which enables free-form content matching with pre-trained language models [25]. This is ideal to support different phrasings from the tutorials that a user might describe. We serve each step title and its text instructions as a pair of question and its response. When user provides a voice command outside of the predefined action list (e.g., "*Show the step of router connection*" in Figure 3e-1), we use the Semantic Reactor to anchor the step that is the highest ranked with similar semantic meaning (e.g., Step 3, "*Connect the router*".)

5.5 Implementation Details

To retrieve the document structure, we use a Markdown renderer developed by our organization that takes an input document of the GitHub Flavored Markdown format [35], similar to common open-sourced tools [27]. For *text instructions*, we synthesize the voiceover using Google's Text-to-Speech API [24] and set a constant synthesis speech speed at 0.5 (i.e., neutral speed rate). For the narration purposes, we remove parentheses in text instructions and include pauses between sentences (d^{pause} =0.35 seconds learned from voiceover studies). Our engine automatically retrieves the Markdown document and its multimedia assets through the tutorial platforms' API, similar to prior work [8]. Our pipeline processes images, videos, and audio files and renders these multimedia materials into a MP4 video based on open source frameworks including MediaPipe [23]. Finally, the Web-based UI is developed using standard HTML5 and JavaScript for video control and the Web Speech API [16] for speech input and output. The automatic editing decisions were saved as metadata and can be edited by a human editor for video re-rendering.

6 RESULTS

To examine the generality of HowToCut and the quality of the generated videos, we created a dataset of 40 existing web tutorials and describe the examples of video outputs from our pipeline.

6.1 Dataset

We selected 40 Markdown-formatted tutorials in 12 categories from the formative dataset we analyzed. These are real-world, publicly available examples, including tutorials from wikiHow.com (shared with permission from wikiHow) and other sources. We selected one to four tutorials from each category based on the following criteria: (1) The step structure is clearly annotated by the Markdown language and follows the tutorial platform's guidelines. (2) The steps are presented in an linear order. (3) Each step contains at least a title and an image or video clip. We filtered articles that have a talking head or audio in any step video, which are not common in popular platforms serving tutorial documents.

6.2 Method and Results

We performed the end-to-end pipeline using HowToCut and were able to create both overview videos and full step videos for all the 40 web tutorials. For each tutorial, HowToCut took an average of 6 minutes to generate the TTS audio files of the same male voice and speech rate (0.5), and 5 minutes for planning and rendering one video. In addition to creating the *default* videos using the algorithm described in Section 5, we also generated three other video versions via naïve approaches for comparison: a *complete* video that narrates all text instructions and is similar to a screen reader, a *compact* video that narrates only the step titles, and an *introduction* video that narrates only the introduction section while concatenating the step images or videos altogether.

Table 2 shows an analysis of the source documents HowToCut captured and the output details with our dataset. In total, our pipeline captured 406 steps that are composed of 23,371 words, 129 images, and 252 videos (8.13-second long on average) from the 40 tutorials. HowToCut added 461 connection sentences to the content, making it a total of 1,968 voiceover sentences saved as individual audio files. On average, each tutorial has 10.48 steps and includes 584.28 words, supported by 3.23 images and 6.30 videos. The generated default video is 82.45 seconds long (with 189 narrated words), two minutes shorter than a complete version of 215.08-second long that is similar to a screen reader.

Figure 7 shows a sample set of videos created by HowToCut. Overall, the automatically-generated videos followed the design guidelines we derived: HowToCut presents concise instructions in a video by organizing the step-by-step procedures in an linear order from the source document. The voiceover, the text overlays, and the visuals are aligned to describe each step in all the videos.

We observed that the voiceover effectively illustrates the step details, so that viewers can focus on the visual actions shown in the video *at the same time*. This is different from reading a web tutorial that requires a learner focusing on either reading the text or watching the step video. In addition, the text overlays of step titles provides additional visual guidance. HowToCut makes consistent decisions on both the audio and visual timing. The voiceover has a reasonable pace with short pauses in between, unlike conventional screen readers that narrate a document continuously. All the videos contain a certain amount of edits, including subtitles, looping, and cuts. For a long voiceover, HowToCut effectively zooms to the region of interest for either an image or a step video (see Figure 7c and d).

7 USER EVALUATION

To evaluate the generated videos by HowToCut and understand how they further support tutorial following, we conducted two user studies. Study I investigated the video quality, including a remote interview study with 12 participants who are DIY enthusiasts and an online survey with 93 respondents. Study II investigated how users followed an automatically-generated video tutorial with navigation control through our UI.

7.1 Study I-1: Viewer Inspection

In the first study, our goals were to (1) understand if tutorial viewers would accept the video quality of an automatically-generated video and (2) verify if the automatic content selection was adequate. We hypothesized that users would find HowToCut's videos reasonable and show a preference toward a shorter video that contains necessary instructions. To answer these assumptions, we designed an interview study with 12 participants from our organization.

7.1.1 Study Design. We sent a study invitation to multiple special interest groups of DIY enthusiasts, including crafting, hardware and wood workshops, and others. There were over 10,000 recipients of our listservs. We received 84 sign-ups through the screening survey. We recruited 12 participants (6 females) who had recently reviewed at least 10 tutorials (either a print copy, a web document, or a video) in the past three months. While we were not able to record and report the age or ethnicity information, participants were professionals of different roles in software industry and all aged over 21. Participants self-reported their familiarity in following tutorials as a Median of 5 (Very Familiar) and creating instructions as $M = 4$ (Familiar). The expertise in video production was $M = 3$ (Neutral).

Table 2: HowToCut converted step-by-step instructions from 40 web tutorials of 12 categories into 160 videos of various video lengths (in seconds), each presents different levels of instructional details.

	Source Document					TTS Voiceover			Output video duration (*d* in seconds) & # of words (*w*)							
	# of steps	# of images	# of videos	Video duration	# of words	# of added sentences	# of audios	wps	*Default*		*Complete*		*Compact*		*Introduction*	
									d	*w*	*d*	*w*	*d*	*w*	*d*	*w*
AVE	10.48	3.23	6.30	8.13	584.28	11.53	49.20	3.2	82.45	189	215.08	570	72.64	102	54.81	85
MIN	5	0	0	1.23	221	6	22	2.8	26.79	52	85.82	211	19.03	25	18.01	45
MAX	25	12	15	44.28	1872	26	130	3.6	204.37	722	664.10	1860	288.02	468	196.28	135
Total	419	129	252	2049.66	23371	461	1968	-	3628	8484	9248	24504	3124	4370	2302	3586

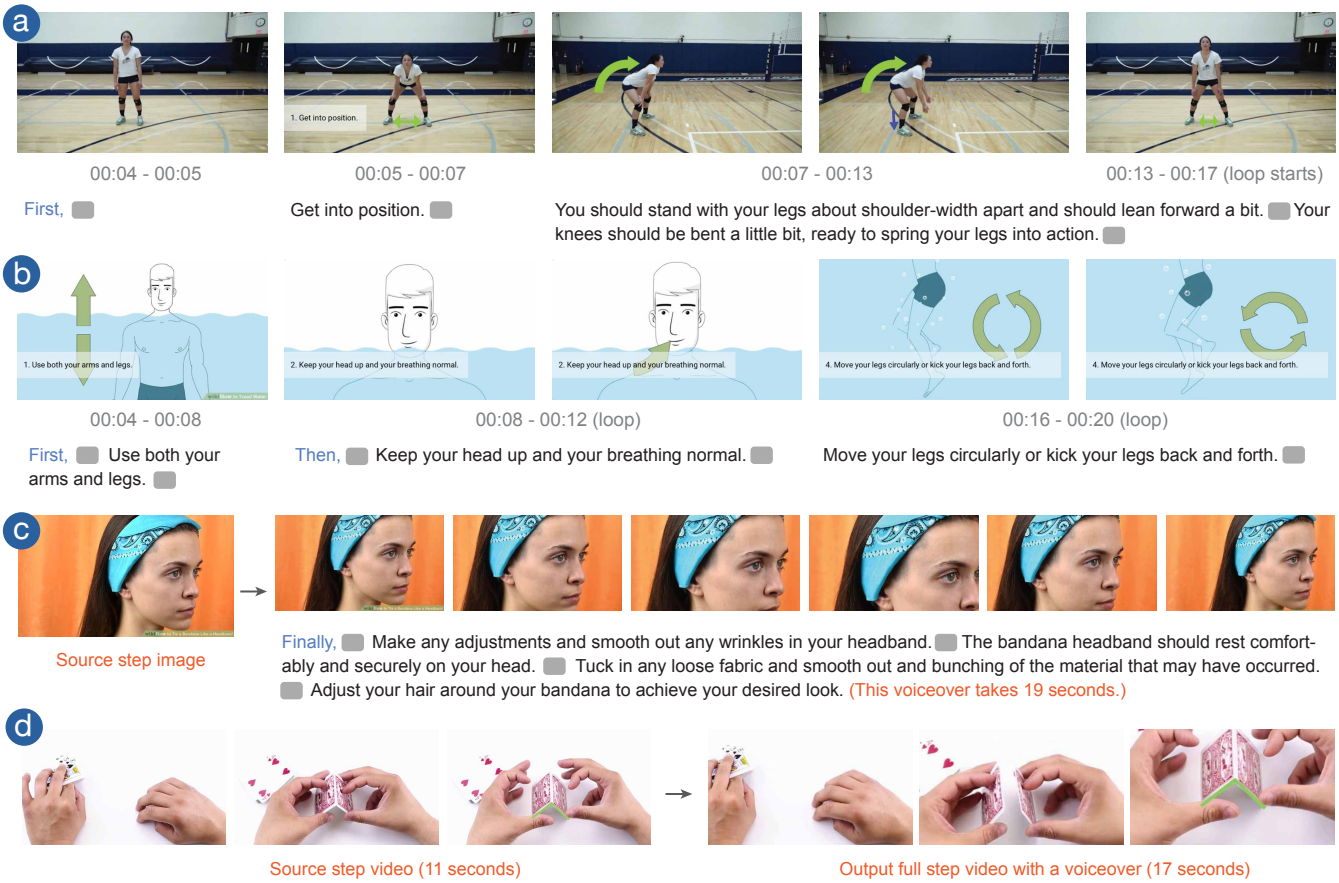

Figure 7: Example automatically-edited videos by HowToCut. A voiceover illustrates sport actions in a video (a) and in an illustrated video (b). For a long voiceover, HowToCut moves the camera view for an image (c) or a step video (d) to dynamically focus on the region of interest. ***Tutorial source from (a) to (d): wikiHow, "How to Bump a Volleyball," "How to Tread Water," "How to Tie a Bandana Like a Headband," "How to Build a Tower of Cards," shared with permission from wikiHow.***

Each participant received a $25 gift card for their participation in a 60-minute remote session.

Materials. We selected four tutorials from our dataset of four categories, including: How to Make Apple Juice (denoted as **T1**), Do String Art (**T2**), Bump a Volleyball (**T3**), and Make Gummies (**T4**). The tutorials were from the same source (wikiHow) in order to provide a consistent instructional style and content quality for reviews. For each tutorial, we created four videos (i.e., four conditions) as described in Section 6.

Procedure. Each session started with video evaluation, followed by tutorial annotation, content review, and a questionnaire. First, participants were asked to review four videos from different tutorials one-by-one. We counterbalanced the tutorials and conditions so that each participant reviewed only one video from **T1** to **T4**, each of a unique version. After each video review, we asked participants

Table 3: Tutorials we provided in our interview studies (Study I-1 with T1-T4 and Study II with S1-S2) and online survey (Study I-2 with R1-R5). Video versions that were not shown to the participants were marked in gray in the table.

Id		Tutorial title	Category	# of Steps	# of Assets				Asset Type	Output video duration (*d* in seconds) & # of words (*w*)							
					Images	Videos	Sentences	Words		*Default*		*Complete*		*Compact*		*Introduction*	
										d	*w*	*d*	*w*	*d*	*w*	*d*	*w*
Warmup		How to Make Hand Sanitizer	Health	4	1	3	24	239	Real scenes	49	106	96	229	40	31	34	80
T1		How to Make Apple Juice	Cooking	8	1	7	40	419		81	189	151	407	61	74	53	76
T2		How to Do String Art	Crafting	10	2	8	52	556		101	241	203	542	69	89	55	45
T3	R1	How to Bump a Volleyball	Sports	8	0	8	50	725		59	141	209	617	38	59	33	92
T4	R2	How to Make Gummies	Health	9	4	5	46	491		65	142	167	412	60	91	37	81
	R3	How to Make Crumpets	Cooking	12	0	12	61	607		134	329	231	589	97	111	95	79
	R4	How to Tread Water for Beginners	Sports	12	4	8	63	638	Illustration	77	172	215	558	54	94	31	86
S1	R5	How to Tie a Bandana Headband	Fashion	9	4	5	48	623	Real scenes	71	148	196	555	64	109	40	70
S2		How to Build a Tower of Cards	Hobby	7	1	6	25	247		75	101	234	110	70	73	60	66

to answer five 5-point Likert-scale questions in terms of the easiness to follow (**Q1**), if they understand the instructions (**Q2**), the pace (**Q3**), the length (**Q4**), and the amount of content (**Q5**) in the video. The scale is ranged from Strongly Disagree (1) to Strongly Agree (5) to each question. For **Q3** to **Q5**, if participants' response was 3 or below, we asked a follow-up question: if the video was too fast or slow (**Q3**), too long or short (**Q4**), or has too much or little content (**Q5**), or if they were unsure. All 12 participants experienced each tutorial and each version.

Once participants finished reviewing four videos, we provided the source tutorial content of the last video they saw. We extracted the text and image or video asset from each step into a slide deck. Participants were asked to highlight the full or partial sentences that they would include in a How-To video as if they were the tutorial author. They were asked not to edit the text instructions at the word level. Next, we introduced a preliminary review UI showing the editing decision by HowToCut of the same tutorial and had participants inspect the details.

7.1.2 Results. Overall, participants found the automatic video editing decisions reasonable ($M = 4$) and commented on HowToCut's advantages as, "*It automates the bulk of the work to create video tutorials*" (P10), "*Helps leverage existing content to reach a wider audience without having to invest excessive editing time*" (P7), "*It greatly simplifies the process of turning a written tutorial into a video, eliminating the need for learning and using complicated video-editing software.*" (P2), "*It provides a relatively painless way to transition from web and text to video*" (P5), and "*it would save a lot of time and encourage publication through multiple media outlets*" (P1). Below we discussed the findings on video quality.

Instructions were easy to follow. Table 4 shows participants' feedback on HowToCut's video quality. All participants found HowToCut's default videos easy to follow (**Q1**) and could understand the instructions (**Q2**) (both $M = 5$). P4 explained, "*The voiceover helped me focus on seeing the volleyball movements in the video, much better than reading text.*" HowToCut's structural alignment ensures the voiceover matches the step visuals. P2 commented, "*It makes it super easy to line up the right words with the right visuals, and understand the length of each part.*" To participants, it was highly noticeable if the text instructions did not match the images or videos, where the *introduction* video condition failed.

Pacing and video length were adequate. Participants agreed that the pace of our default videos was right ($M = 4$ to **Q3**), especially the voiceover, as noted by P9 who is an expert on accessibility technologies. On the contrary, the naïve approaches failed: Narrating every text instruction from a written tutorial was rated slow (the *complete* video), while narrating only the step titles is too fast (the *compact* video). The responses to the amount of content (**Q3**) reflected similar ratings. The length of HowToCut's videos received the highest rating ($M = 4.5$ to **Q4**).

Automatic editing decisions were reasonable. Participants strongly agreed that the concept of automatically converting a step-by-step tutorial to a video was straightforward ($M = 5$). They pointed out that composing a blog post or a document is typically faster than editing a video, but an article can include too many details that are not suitable for a voiceover. Take **T3** (volleyball) as an example. It is especially difficult to include the detailed verbal descriptions when a video best illustrates a sport activity. In the exercise of selecting text instructions from the source tutorial into a video script, all of the participants selected sentences from paragraphs by removing less critical information, such as openings ("when it comes to"), assumptions ("if you are looking for (...)"), or tips. P5 commented, he took a How-To video as a starting point, where details can be listed in supplementary materials or in a document. Therefore, it's best to make a video right to the point and exclude supplementary instructions.

7.2 Study I-2: Audience Survey

To understand how general audience perceives the video quality, we conducted an online survey with two conditions, HowToCut's default videos and the complete version as the baseline. We hypothesized that our default videos are easier to consume as it is more concise than a fully narrated version, which is similar to a conventional screen reader. We selected five tutorials from our results of different topics (see **R1** to **R5** in Table 3). Each survey included one random video from **R1** to **R5** of the two conditions. The video was played once, followed by the same list of five Likert-scale questions as used in Study I-1, an additional question about viewer's familiarity with the topic of the video (**Q6**), and an optional text field for feedback. We distributed our online survey via multiple internal listservs for a wider range of audience that included professionals

Table 4: Participants' responses to five video-quality questions at 5-point Likert scale in our Study I. Each response was recorded after reviewing one of the 16 videos automatically generated by HowToCut using four editing strategies.

	Q1 (easy to follow)			*Q2 (understand)*			*Q3 (pace is right)*			*Q4 (length is right)*			*Q5 (amount is right)*		
	MIN	MED	MAX	MIN	MED	MAX	MIN	MED	MAX	MIN	MED	MAX	MIN	MED	MAX
Default	3	5	5	3	5	5	2	4	5	3	4.5	5	2	4	5
Complete	1	4	5	1	4	5	1	3	5	2	4	5	1	3	5
Compact	2	4.5	5	1	5	5	2	4	5	2	4	5	1	3.5	5
Introduction	1	3	5	1	3	5	1	2.5	5	1	4	5	1	2	4

of all roles in our organization. Participants did not receive any monetary incentive for survey completion.

We received 93 unique responses to the online survey. Each condition was reviewed by at least 43 participants, and each tutorial was reviewed by at least 22 participants due to randomization. We analyzed the data based on bootstrap method [40]. Similar to Study I-1 with DIY expert, participants from general audiences agreed that the videos were easy to follow (*ave*=4.16 and 3.59 to **Q1** of the default videos and the baseline respectively). They could understand the instructions (*ave*=4.385 and 4.198 to **Q2**) and were positive about the pace (**Q3**), the video length (**Q4**), and the amount of content (**Q5**) with ratings over 3. Survey responses from participants indicated no statistically significant difference between the two conditions to our hypothesis. Although we were not able to conclude if the audience preferred either condition, participants were positive in the video quality and provided encouraging comments: "*it doesn't look like an automatically generated video.*", "*if this is a machine generated how-to video from a written recipe, that would be pretty cool!*", and "*it was very simple, easy-to-understand, and efficient!*"

7.3 Study II: Tutorial Following

To further investigate if our automatically-generated videos with navigation control could help users follow a task, we conducted an within-subject remote interview study with 8 DIY enthusiasts.

7.3.1 Study Design. Similar to Study I, we sent an invitation to internal interest groups of over 10,000 recipients. We received 40 sign-ups and selected 8 participants (4 females) of various professionals who did not join our prior studies. All of them had followed at least five tutorials in the past three months. We provided the same amount of incentive ($25) as Study I-1 for an 60-minute session.

Materials. We selected two tutorials that utilize objects that are widely available at home but require unique skills and can be completed on a tabletop, including (**S1**) How to Tie a Bandana Like a Headband, which uses a bandana or a towel, and (**S2**) How to Build a Tower of Cards, which uses a deck of playing cards. These tutorials were from the same source as Study I (wikiHow) to provide consistent quality for comparison (see Table 3).

Conditions. We provided two conditions, a step-by-step document as a baseline and our UI. We removed advertising and hyperlinks from the source tutorials and presented all steps in an linear, top-down format as the baseline. Each step shows the step image or looped video, next to its step number, title, and supportive text instructions. The control condition provides HowToCut's UI with the same functionality as shown in Figure 3, except for the user voice input that we chose not to evaluate in the remote setting.

Procedure. We counterbalanced the tutorials and conditions. For each condition, we started with a warm-up task to help participants familiarize with the UI. We then provided the link to the tutorial task and asked them to complete within 10 minutes. Participants could choose to end the task early after at least two attempts of the steps. We logged the UI interaction traces. Then, we asked participants to verbally answer five 5-point Likert-scale questions in terms of the tutorial quality, components, and their preferences. After completing both tasks and ratings, we asked their thoughts on whether the HowToCut video was human edited at the scale from 1 to 10. Each session ended with a discussion and a questionnaire.

7.3.2 Results. All the participants walked through the tutorials of both formats and provided valuable feedback. They commented the advantages of HowToCut as: "*brings the clarity of the tutorial steps*" (P1), "*the ability to choose the viewing format*" (P4), "*Step wise distinctions of the video, easier to navigate*" (P7), and "*I'm a visual learner, so it's nice being able to see everything step by step visually*" (P8). Below we summarized our findings.

HowToCut supported task following. Participants rated tutorials of both conditions useful and easy to follow (both $M = 5$). When being asked if they would use this tool again to follow a tutorial of any topic, all strongly agreed ($M = 5$) in the HowToCut condition, compared to the baseline $M = 4$. Participants provided the same rating in the favor of HowToCut over conventional video platforms ($M = 5$) and static tutorials ($M = 5$).

We observed strategies across participants to review and follow tutorials using different formats. In the *baseline* condition, participants all glanced through the document from the first step to the end. Then, they scrolled back to Step 1 and started performing the steps one by one. Half of them read the text aloud. In the *HowToCut* condition, all participants played the videos from the beginning to the end. Six of them directly performed the task when watching. For the task **S1**, two of the four participants completed the task along with the one-minute video without replay. The other two participants clicked one to five steps to confirm their progress. For **S2**, all anchored back to the step of their current progress after the video ended. They reviewed the specific step, followed instructions, and proceeded to replay the following steps one by one.

With the continuous video and a voiceover, HowToCut allowed participants to focus on task following. Participants explained that for complicated tasks (such as **S2**), HowToCut's UI was useful as it "*allows me to know where I'm at when I came back to the computer after finishing that step*" (P1). We did not observe significant differences of task completion time between the two conditions.

Voiceover played an important role. Participants found the voiceover helpful ($M = 4$). P8 specifically commented that "*it slows*

me down to digest everything necessary" where the voiceover reminded him to "*place about 1cm of space between them (the cards)*", which he might have missed from reading the text. Similar to Study I-1, the voiceover allowed participants to focus on the video frames to see the actions. When following a task, the voiceover further allowed them to receive instructions while working on the project. These findings align with prior studies that suggested the importance of narrated tutorials [50], step hierarchy [48], and voice control [6]. In addition, we observed that a long voiceover break (i.e., $d_k^{visual} \geq d_k'^{VO}$) led participants to pause the video. The final step of **S2** showed a 15-second video segment, but the voiceover only covered 2 seconds ("*Finally, finished,*") as the source tutorial did not contain more text instructions. P5 suggested adding background music, which is a common practice for instructional videos.

Other remarks. Participants agreed that it is best to avoid including every detail from a document in the video, which aligns with what we observed in the formative video study (see Section 3). A tutorial should provide an overview of the task, while providing easy-to-access tips when learners need more information. For example, with step 2 in **S2**, all the participants went back to the tutorial after one to more attempts in both conditions. It is important to allow viewers to have full control of the instructional details. HowToCut's UI helped participants anchor back to the step and reveal the full video or text instructions.

Finally, participants supposed that the HowToCut video was human-edited ($M = 7.5$ at the 10-point scale, where six participants rated above 6.) P1 explained, "*I thought the video was from YouTube! I was so focused on the task and did not notice anything mechanic. The voiceover was quite good.*" P6 commented, "*I'm surprised by the quality if this is automatically generated. I'd surely want to use this tool again.*"

8 DISCUSSION AND OPPORTUNITIES

Overall, we received positive feedback on video quality and presentation from study participants. DIY enthusiasts found HowToCut useful to capture the instructions and follow a task. General audience found HowToCut's videos easy to watch and could understand the instructions, even to an unfamiliar topic. Below we describe the opportunities of our approach for instructional video creation from a document.

Quality of the source document. Our current pipeline relies on a quality source tutorial. It requires an annotated step-by-step structure, sufficient text and visual materials, and clear text instructions. Although we mainly present content from wikiHow in this paper thanks to wikiHow's permission, we have tested tutorials from other platforms and observed similar results. However, we acknowledged limitations of our approach. For a step that lacks of visual assets, HowToCut does not automatically fill the visual gap using stock assets or supporting materials linked from a tutorial. In addition, when a step video contains multiple shots, HowToCut does not align the voiceover with the exact actions or objects as human editors often optimize. We suggest that further integration could improve the tutorial presentation.

Navigation control and personalization. HowToCut provides a step-based video navigation that could enhance existing video platforms. Participants looked for more video controls, including playback speed adjustment (Study II-P2) and voiceover characters (Study I-P10) given the tutorial context. Given different goals and context, tutorial followers may prefer more personalized experiences that reveal content in a format based on their needs or experiences. We suggest future research considering user expertise and preferences for interactive tutorial consumption.

Support international languages. Finally, we look forward to extending HowToCut to multiple languages to reach more audience. Existing tutorial platforms have devoted increasing efforts in content translation [53]. Recent advancement in translation techniques also makes it possible to synthesize a document in another language [28]. As translation can lead to a voiceover of different lengths, we suggest that our approach of automatic video editing is flexible to handle such an input. All in all, we aim to support people of different learning preferences and needs and make information accessible and useful.

9 CONCLUSION

In this paper, we describe an automatic approach that converts a Markdown-formatted multimedia tutorial to instructional videos, which present a step-by-step procedure with synthesized voiceover and images or videos. Our method extracts the structural content in a document based on its Markdown annotation. Our pipeline makes automatic video editing decisions that consider the information load. It selects and converts text instructions into a voiceover, with a timed presentation of images, video clips, and text overlays. Using Computer Vision techniques, we reframe videos to allow detailed verbal instructions. We provide a user interface for tutorial followers to navigate the instructional content via GUI and voice input. We describe common principles and editing strategies learned from an analysis of 125 web tutorials and their videos that we confirmed with professional video producers. Through studies with 20 participants and an online survey with 93 responses, we evaluated our automatically-generated videos and found that our method was able to effectively create informative videos from a tutorial document for task review and completion.

ACKNOWLEDGMENTS

The tutorials from wikiHow are shared with permission. We thank Sonia Chang and Elizabeth Douglas at wikiHow Inc. for their support. We thank all the participants in our studies for their feedback to move this research forward. In addition, this work has been possible thanks to the support of people including, but not limited to the following (in alphabetical order of last name): Eric Gagneraud, Nick Monroe, Bo Pang, Mogan Shieh, Rachel Soh, Radu Soricut, Zheng Sun, Janel Thamkul, and Aparna Warrier.

REFERENCES

[1] Maneesh Agrawala, Doantam Phan, Julie Heiser, John Haymaker, Jeff Klingner, Pat Hanrahan, and Barbara Tversky. 2003. Designing Effective Step-by-step Assembly Instructions. *ACM Trans. Graph.* 22, 3 (July 2003), 828–837. https://doi.org/10.1145/882262.882352

[2] Faisal Ahmed, Yevgen Borodin, Andrii Soviak, Muhammad Islam, I.V. Ramakrishnan, and Terri Hedgpeth. 2012. Accessible Skimming: Faster Screen Reading of Web Pages. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. Association for Computing Machinery, New York, NY, USA, 367–378. https://doi.org/10.1145/2380116.2380164

[3] Floraine Berthouzoz, Wilmot Li, and Maneesh Agrawala. 2012. Tools for Placing Cuts and Transitions in Interview Video. *ACM Trans. Graph.* 31, 4, Article Article

67 (July 2012), 8 pages. https://doi.org/10.1145/2185520.2185563

[4] Juan Casares, A. Chris Long, Brad A. Myers, Rishi Bhatnagar, Scott M. Stevens, Laura Dabbish, Dan Yocum, and Albert Corbett. 2002. Simplifying Video Editing Using Metadata. In *Proceedings of the 4th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques (DIS '02)*. Association for Computing Machinery, New York, NY, USA, 157–166. https://doi.org/10.1145/778712.778737

[5] Minsuk Chang, Mina Huh, and Juho Kim. 2021. RubySlippers: Supporting Content-Based Voice Navigation for How-to Videos. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 97, 14 pages. https://doi.org/10.1145/3411764.3445131

[6] Minsuk Chang, Anh Truong, Oliver Wang, Maneesh Agrawala, and Juho Kim. 2019. How to Design Voice Based Navigation for How-To Videos. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, New York, NY, USA, Article 701, 11 pages. https://doi.org/10.1145/3290605.3300931

[7] Jiajian Chen, Jun Xiao, and Yuli Gao. 2010. ISlideShow: A Content-Aware Slideshow System. In *Proceedings of the 15th International Conference on Intelligent User Interfaces (IUI '10)*. Association for Computing Machinery, New York, NY, USA, 293–296. https://doi.org/10.1145/1719970.1720014

[8] Peggy Chi, Zheng Sun, Katrina Panovich, and Irfan Essa. 2020. Automatic Video Creation From a Web Page. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. Association for Computing Machinery, New York, NY, USA, 279–292. https://doi.org/10.1145/3379337.3415814

[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 (UIST '12)*. ACM, New York, NY, USA, 93–102. https://doi.org/10.1145/2380116.2380130

[10] Pei-Yu Chi, Joyce Liu, Jason Linder, Mira Dontcheva, Wilmot Li, and Björn Hartmann. 2013. DemoCut: Generating Concise Instructional Videos for Physical Demonstrations. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST '13)*. Association for Computing Machinery, New York, NY, USA, 141–150. https://doi.org/10.1145/2501988.2502052

[11] Ohad Fried, Ayush Tewari, Michael Zollhöfer, Adam Finkelstein, Eli Shechtman, Dan B Goldman, Kyle Genova, Zeyu Jin, Christian Theobalt, and Maneesh Agrawala. 2019. Text-Based Editing of Talking-Head Video. *ACM Trans. Graph.* 38, 4, Article Article 68 (July 2019), 14 pages. https://doi.org/10.1145/3306346.3323028

[12] Google. 2020. *AutoFlip: Saliency-aware Video Cropping*. Retrieved April, 2021 from https://google.github.io/mediapipe/solutions/autoflip

[13] Google. 2020. *Face Detection - mediapipe*. Retrieved April, 2021 from https://google.github.io/mediapipe/solutions/face_detection

[14] Google. 2020. *Object Detection - mediapipe*. Retrieved April, 2021 from https://google.github.io/mediapipe/solutions/object_detection

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

[16] Web Platform Incubator Community Group. 2020. *Web Speech API*. Retrieved April, 2021 from https://wicg.github.io/speech-api/

[17] John Gruber. 2004. Daring fireball: Markdown. (2004). https://daringfireball.net/projects/markdown/

[18] Joshua M. Hailpern and Bernardo A. Huberman. 2014. Odin: Contextual Document Opinions on the Go. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. Association for Computing Machinery, New York, NY, USA, 1525–1534. https://doi.org/10.1145/2556288.2556959

[19] Bernd Huber, Hijung Valentina Shin, Bryan Russell, Oliver Wang, and Gautham J. Mysore. 2019. B-Script: Transcript-Based B-Roll Video Editing with Recommendations. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, Article Paper 81, 11 pages. https://doi.org/10.1145/3290605.3300311

[20] iFixit. 2021. *About iFixit*. Retrieved July, 2021 from https://www.ifixit.com/Info/

[21] iFixit. 2021. *Wiki Formatting And Syntax*. Retrieved April, 2021 from https://www.ifixit.com/Help/Wiki_Formatting_And_Syntax

[22] Corneliu Ilisescu, Halil Aytac Kanaci, Matteo Romagnoli, Neill D. F. Campbell, and Gabriel J. Brostow. 2017. *Responsive Action-Based Video Synthesis*. Association for Computing Machinery, New York, NY, USA, 6569–6580. https://doi.org/10.1145/3025453.3025880

[23] Google Inc. 2020. *MediaPipe: a cross-platform framework for building multimodal applied machine learning pipelines*. Retrieved March, 2021 from https://github.com/google/mediapipe/

[24] Google Inc. 2020. *Text-to-Speech: Lifelike Speech Synthesis*. Retrieved September, 2020 from https://cloud.google.com/text-to-speech/

[25] Google Inc. 2021. *Semantic Reactor: Experiment with machine learning language models*. Retrieved March, 2021 from https://research.google.com/semanticexperiences/semantic-reactor.html

[26] Instructables. 2021. *Share what you make on Instructables!* Retrieved April, 2021 from https://www.instructables.com/create/

[27] Christopher Jeffrey. 2018. *Marked: A markdown parser and compiler. Built for speed*. Retrieved April, 2021 from https://github.com/markedjs/marked

[28] Ye Jia, Ron J. Weiss, Fadi Biadsy, Wolfgang Macherey, Melvin Johnson, Zhifeng Chen, and Yonghui Wu. 2019. Direct speech-to-speech translation with a sequence-to-sequence model. In *InterSpeech '19*. arXiv:cs.CL/1904.06037

[29] Murat Kalender, Mustafa Eren, Zonghuan Wu, Ozgun Cirakman, Sezer Kutluk, Gunay Gultekin, and Emin Korkmaz. 2018. Videolization: knowledge graph based automated video generation from web content. *Multimedia Tools and Applications* 77 (12 2018). https://doi.org/10.1007/s11042-016-4275-4

[30] Kyoungkook Kang and Sunghyun Cho. 2019. Interactive and Automatic Navigation for 360° Video Playback. *ACM Trans. Graph.* 38, 4, Article 108 (July 2019), 11 pages. https://doi.org/10.1145/3306346.3323046

[31] 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 (CHI '14)*. Association for Computing Machinery, New York, NY, USA, 4017–4026. https://doi.org/10.1145/2556288.2556986

[32] Mackenzie Leake, Abe Davis, Anh Truong, and Maneesh Agrawala. 2017. Computational Video Editing for Dialogue-Driven Scenes. *ACM Trans. Graph.* 36, 4, Article Article 130 (July 2017), 14 pages. https://doi.org/10.1145/3072959.3073653

[33] Mackenzie Leake, Hijung Valentina Shin, Joy O. Kim, and Maneesh Agrawala. 2020. Generating Audio-Visual Slideshows from Text Articles Using Word Concreteness. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3313831.3376519

[34] Sean J. Liu, Maneesh Agrawala, Stephen DiVerdi, and Aaron Hertzmann. 2019. View-Dependent Video Textures for 360° Video. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*. Association for Computing Machinery, New York, NY, USA, 249–262. https://doi.org/10.1145/3332165.3347887

[35] John MacFarlane. 2019. *GitHub Flavored Markdown Spec*. Retrieved September, 2020 from https://github.github.com/gfm/

[36] Peter Mohr, Bernhard Kerbl, Michael Donoser, Dieter Schmalstieg, and Denis Kalkofen. 2015. *Retargeting Technical Documentation to Augmented Reality*. Association for Computing Machinery, New York, NY, USA, 3337–3346. https://doi.org/10.1145/2702123.2702490

[37] Peter Mohr, David Mandl, Markus Tatzgern, Eduardo Veas, Dieter Schmalstieg, and Denis Kalkofen. 2017. *Retargeting Video Tutorials Showing Tools With Surface Contact to Augmented Reality*. Association for Computing Machinery, New York, NY, USA, 6547–6558. https://doi.org/10.1145/3025453.3025688

[38] Alok Mysore and Philip J. Guo. 2017. Torta: Generating Mixed-Media GUI and Command-Line App Tutorials Using Operating-System-Wide Activity Tracing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. Association for Computing Machinery, New York, NY, USA, 703–714. https://doi.org/10.1145/3126594.3126628

[39] Amy Pavel, Gabriel Reyes, and Jeffrey P. Bigham. 2020. Rescribe: Authoring and Automatically Editing Audio Descriptions. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. Association for Computing Machinery, New York, NY, USA, 747–759. https://doi.org/10.1145/3379337.3415864

[40] Abhijit Pol and Christopher Jermaine. 2005. Relational Confidence Bounds Are Easy with the Bootstrap. In *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data (SIGMOD '05)*. Association for Computing Machinery, New York, NY, USA, 587–598. https://doi.org/10.1145/1066157.1066224

[41] Consumer Specialists. 2020. *How COVID-19 Is Reshaping The Home Improvement Market*. Retrieved July, 2020 from http://consumerspecialists.com/welcome/new-research/

[42] Sarah Taylor, Taehwan Kim, Yisong Yue, Moshe Mahler, James Krahe, Anastasio Garcia Rodriguez, Jessica Hodgins, and Iain Matthews. 2017. A Deep Learning Approach for Generalized Speech Animation. *ACM Trans. Graph.* 36, 4, Article 93 (July 2017), 11 pages. https://doi.org/10.1145/3072959.3073699

[43] Maartje ter Hoeve, Robert Sim, Elnaz Nouri, Adam Fourney, Maarten de Rijke, and Ryen W. White. 2020. Conversations with Documents: An Exploration of Document-Centered Assistance. *Proceedings of the 2020 Conference on Human Information Interaction and Retrieval* (Mar 2020). https://doi.org/10.1145/3343413.3377971

[44] Cristen Torrey, Elizabeth F. Churchill, and David W. McDonald. 2009. Learning How: The Search for Craft Knowledge on the Internet. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. Association for Computing Machinery, New York, NY, USA, 1371–1380. https://doi.org/10.1145/1518701.1518908

[45] Cristen Torrey, David W McDonald, Bill N Schilit, and Sara Bly. 2007. How-To pages: Informal systems of expertise sharing. In *ECSCW 2007*. Springer, 391–410.

[46] Anh Truong, Floraine Berthouzoz, Wilmot Li, and Maneesh Agrawala. 2016. QuickCut: An Interactive Tool for Editing Narrated Video. In *Proceedings of the*

29th Annual Symposium on User Interface Software and Technology (UIST '16). Association for Computing Machinery, New York, NY, USA, 497–507. https://doi.org/10.1145/2984511.2984569

[47] Anh Truong, Sara Chen, Ersin Yumer, David Salesin, and Wilmot Li. 2018. Extracting Regular FOV Shots from 360 Event Footage. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, New York, NY, USA, Article Paper 316, 11 pages. https://doi.org/10.1145/3173574.3173890

[48] Anh Truong, Peggy Chi, David Salesin, Irfan Essa, and Maneesh Agrawala. 2021. Automatic Generation of Two-Level Hierarchical Tutorials from Instructional Makeup Videos. In *Proceedings of the 2021 ACM Conference on Human Factors in Computing Systems (CHI '21)*.

[49] Tiffany Tseng and Mitchel Resnick. 2014. Product versus Process: Representing and Appropriating DIY Projects Online. In *Proceedings of the 2014 Conference on Designing Interactive Systems (DIS '14)*. Association for Computing Machinery, New York, NY, USA, 425–428. https://doi.org/10.1145/2598510.2598540

[50] Sylvaine Tuncer, Barry Brown, and Oskar Lindwall. 2020. On Pause: How Online Instructional Videos Are Used to Achieve Practical Tasks. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376759

[51] Miao Wang, Guo-Wei Yang, Shi-Min Hu, Shing-Tung Yau, and Ariel Shamir. 2019. Write-a-Video: Computational Video Montage from Themed Text. *ACM Trans. Graph.* 38, 6, Article 177 (Nov. 2019), 13 pages. https://doi.org/10.1145/3355089.3356520

[52] Ryan Weber and Karl Stolley. 2018. *Transitional Devices*. Retrieved July, 2021 from https://www.wheaton.edu/academics/services/writing-center/writing-resources/transitions/

[53] wikiHow. 2019. *wikiHow:Herald/2019 Year in Review*. Retrieved April, 2021 from https://www.wikihow.com/wikiHow:Herald/2019-Year-in-Review

[54] wikiHow. 2021. *How to Format a wikiHow Article*. Retrieved April, 2021 from https://www.wikihow.com/Format-a-wikiHow-Article

[55] Nora S. Willett, Wilmot Li, Jovan Popovic, and Adam Finkelstein. 2017. Triggering Artwork Swaps for Live Animation. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. Association for Computing Machinery, New York, NY, USA, 85–95. https://doi.org/10.1145/3126594.3126596

[56] Haijun Xia, Jennifer Jacobs, and Maneesh Agrawala. 2020. Crosscast: Adding Visuals to Audio Travel Podcasts. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. Association for Computing Machinery, New York, NY, USA, 735–746. https://doi.org/10.1145/3379337.3415882

[57] Masahiro Yamaguchi, Shohei Mori, Peter Mohr, Markus Tatzgern, Ana Stanescu, Hideo Saito, and Denis Kalkofen. 2020. *Video-Annotated Augmented Reality Assembly Tutorials*. Association for Computing Machinery, New York, NY, USA, 1010–1022. https://doi.org/10.1145/3379337.3415819

[58] Yu Zhong, T. V. Raman, Casey Burkhardt, Fadi Biadsy, and Jeffrey P. Bigham. 2014. JustSpeak: Enabling Universal Voice Control on Android. In *Proceedings of the 11th Web for All Conference (W4A '14)*. ACM, New York, NY, USA, Article 36, 4 pages. https://doi.org/10.1145/2596695.2596720

[59] Qingxiaoyang Zhu and Hao-Chuan Wang. 2021. Is a GIF Worth a Thousand Words? Understanding the Use of Dynamic Graphical Illustrations for Procedural Knowledge Sharing on wikiHow. In *ECSCW '21*.

[60] Douglas E. Zongker and David H. Salesin. 2003. On Creating Animated Presentations. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '03)*. Eurographics Association, Goslar, DEU, 298–308.