# Automatic Video Creation From a Web Page

**Peggy Chi**[1], **Zheng Sun**[1], **Katrina Panovich**[2], **Irfan Essa**[1,3]
[1]Google Research, Mountain View, CA, USA; [2]YouTube, San Bruno, CA, USA
[3]Georgia Institute of Technology, Atlanta, GA, USA
url2video@google.com

## ABSTRACT
Creating marketing videos from scratch can be challenging, especially when designing for multiple platforms with different viewing criteria. We present URL2Video, an automatic approach that converts a web page into a short video given temporal and visual constraints. URL2Video captures quality materials and design styles extracted from a web page, including fonts, colors, and layouts. Using constraint programming, URL2Video's design engine organizes the visual assets into a sequence of shots and renders to a video with user-specified aspect ratio and duration. Creators can review the video composition, modify constraints, and generate video variation through a user interface. We learned the design process from designers and compared our automatically generated results with their creation through interviews and an online survey. The evaluation shows that URL2Video effectively extracted design elements from a web page and supported designers by bootstrapping the video creation process.

### Author Keywords
Video creation; video storyboard; web document; web design; storytelling; creativity tools.

### CCS Concepts
•**Human-centered computing** → **Human computer interaction (HCI);**

## INTRODUCTION
Business owners host a website that illustrates their services. Such web content often contains a consistent theme and visual representations, including logos, marketing text, images, and color choices [30, 28]. While owners design a compelling web presence or campaign, the materials are rarely re-purposed for other multimedia consumption, such as videos, a popular medium for introducing content given its dynamic visuals that make it engaging and easy to consume [29].

Video editing takes significant time, effort, and budget. It requires continuous temporal and spatial decisions of composing visual assets [2]. Creators organize assets–including video footage, images, and text–into a sequence of shots to fill a fixed video duration. For branding, creators consider the visual design of each shot, which commonly involves colors, sizes, and graphical layout. While editing one single video is a time-consuming process that could be partially automated [14, 25, 40], producing *multiple* videos from the same theme can be even more challenging. Recent trends encourage creators to publish different video versions based on the viewing experiences [55]. In 2019, nearly a third of advertisers developed videos for cross-screen purposes [29]. Each platform has its unique optimal criteria, including aspect ratio (wide screen on a desktop and portrait on a mobile device) and video length (varied from few seconds to minutes). It would take significant efforts to re-organize the materials and adjust the layouts of a video while maintaining similar visual styles.

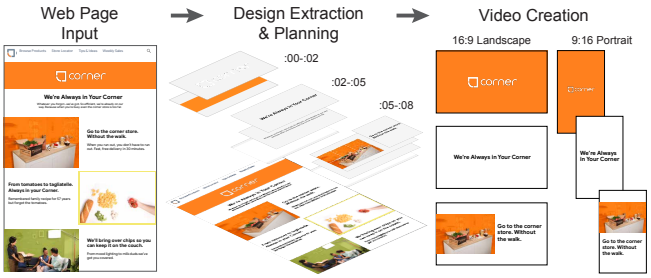


**Figure 1. Given a URL and user-specified parameters, URL2Video automatically generates a marketing video that captures the content and visual design from the source page. It makes both temporal and visual editing decisions to organize web assets to a sequence of shots. Users can examine the design components and modify constraints through our user interface to refine the outputs.**

In this paper, we introduce URL2Video, an automatic approach for converting a web page into a marketing video given user-defined temporal and spatial constraints (see Figure 1). Our generated videos capture visual representations and descriptions extracted from the source web page. We introduce a pipeline that segments a web document to identify quality materials–such as text and images–and their hierarchical structure. URL2Video extracts the visual styles of web components, such as fonts, color themes, and layouts. Its design engine organizes the materials and creates a sequence of shots to fulfill the temporal and spatial constraints. To enable creators to quickly iterate the video variation, we provide a user interface that visualizes the selected assets from the web page and its storyboard (see Figure 3). Users can modify the decision and the output specification, such as aspect ratio and video length.


*UIST '20, October 20–23, 2020, Virtual Event, USA*

ACM ISBN 978-1-4503-7514-6/20/10.
https://doi.org/10.1145/3379337.3415814

We evaluated automatically-generated video results from 50 web pages. We further compared our selected results with creations from designers through interviews with eight designers and a survey with 65 regular viewers. The findings suggested that our pipeline efficiently supports the video creation process. Our work makes the following contributions:

- An automatic approach to generate videos from a web page based on constraint programming to satisfy user-specified parameters of video duration (temporal constraints) and size and aspect ratios (spatial constraints).
- Methods to convert hierarchical assets to a video that maintains a specific visual design.
- An evaluation of automatically generated videos from web pages with professional designers and general audience.

## RELATED WORK

URL2Video builds on prior work of design understanding and computational techniques for video creation. We review and discuss our relationship with the related efforts in these areas.

### Design Understanding

Researchers have proposed computational methods of design understanding and creation for digital media, including web pages, mobile applications, and advertisement. Early work has demonstrated effective techniques for identifying design metrics [35], finding advertising keywords from a web page [63], and retargeting web design [39, 38, 3] based on semantic structures. Recent work further explored UI layout and interaction flow of mobile applications by learning the content hierarchy and semantics [16, 17, 42]. Based on a deep understanding of web page content, layouts can be optimized to help direct viewers' attention [50, 54] or distribute to multiple UI targets [45, 24, 46]. These efforts provide valuable insights for making online content more accessible and useful by learning from document structures and design intents.

There is increasing amount of research on supporting advertisement using computational methods, including modeling audience's visual interests [64] and attention flow [49] or emotion responses to mobile ads [51], and understanding image and video ads [26, 62, 27]. Beyond commercial ads, VisiBlends introduced techniques to combine objects based on their semantic visuals and constraints in graphical design, which can be used to convey a marketing message [15]. In URL2Video we are interested in assisting creators in automatically generating videos from an existing web page. Our work is built upon existing techniques of web and ad understanding, but we focus on converting web content to a short video given both the temporal and spatial constraints with design principles.

### Computational Video Editing

Video creation can be a multi-stage, effort-consuming process. Researchers have suggested a variety of techniques to automate the process of video editing either fully or partially. Video can be summarized by constraints [48, 4, 6] or presented as a storyboard [23]. Editing tools can be tailored for specific domains, such as conversational video [40, 22], interviews [5], physical tasks [14, 57] for particular types of footage [25, 58]. Following these efforts, our work aims to automate the creation process for a specific domain–digital marketing. However, our source materials are static web pages that often do not contain video assets. Therefore, our computational approach focuses on the video composition given a set of design attributes.

Finally, prior work has proposed techniques to automate animation from static materials, including drawing [59, 56, 60], images [61], slideshows [13, 12, 66], and comic books [44, 9]. Common approaches analyze moving traces and apply movements to subjects that are automatically or manually segmented. Recent research [37] and online services have developed methods that transfer a web page (e.g., a Wikipedia page or a blog post) to a video. Existing techniques focus on matching and placing content in a video template. As we are interested in a different problem–converting web assets to marketing videos, we chose to focus on converting graphic design to maintain the visual similarity while satisfying the video constraints, including time and aspect ratio.

## LEARNING FROM DESIGNERS

We describe the terminology and our assumptions from multiple domains that URL2Video touches on, including web page design, video editing, and marketing. To understand existing video creation processes, we invited six designers and observed their design decisions, and discuss how these decisions impact URL2Video's framework.

### Definition

A *web page* is a document composed of structured content written in Hypertext Markup Language (HTML) that can be rendered by a web browser. To obtain the page content with a meaningful semantic structure, we assume that elements are properly annotated by HTML tags, such as `H1` for a heading and `IMG` for an image. An HTML document forms a tree structure, accessible via the Document Object Model (DOM) interface. The multimedia materials contained in a page are available *assets*, including text, images, and videos. We assume that a web page has a minimum degree of visual design specified by Cascading Style Sheets (CSS). We acknowledge that there are emerging Web technologies for advanced user interaction. For now we focus on static web pages and do not support encapsulated Web applications (e.g., Adobe Flash) that require additional parsing mechanism. Pages that heavily involve user input or animations are also beyond the scope of our work.

A *video* is a medium that displays moving visual media within a fixed length of time (i.e., *duration*). Its width to height defines the *aspect ratio* and frame size, such as 1920 by 1080 pixels for a 16:9 landscape 1080p video. A video may contain one or a series of *scenes*, each maintains similar visuals of one or more subjects. A scene may be composed of several *shots*, each runs for an uninterrupted period of time [2]. Although audio is a critical component of videos, we do not generate audible content at this point.

Video production involves multiple stages from planning, assets capturing, editing, to distribution [2]. Prior to capturing and editing footage, *storyboarding* is a popular way to "prototype" a video. It is widely used in conventional filmmaking and has been adopted in advertising for commercial campaigns and corporate video production. We focus on videos for *digital*

**Table 1. We invited six designers, each reviewed a unique homepage and composed a video storyboard using the extracted assets in our formative study. Designers selected various numbers of assets into scenes for a 10-second video. Partial snapshots can be found in Figure 2 (a for p1, b for p5, and c/d for p6). None of the pages has animated content.**

| Page | Category | Video Creation | | # of Assets Used (Total) | | |
|---|---|---|---|---|---|---|
| | | Scene | Shot | Image | Text | Color |
| warm-up | Service | 5 | 5 | 5 (9) | 3 (13) | 3 (4) |
| p1 | Electronics | 5 | 7 | 10 (10) | 5 (16) | 3 (6) |
| p2 | Fashion | 7 | 7 | 7 (32) | 7 (40) | 4 (5) |
| p3 | Food&Drink | 4 | 10 | 5 (6) | 5 (18) | 4 (8) |
| p4 | NPO–Event | 4 | 4 | 3 (12) | 4 (26) | 4 (6) |
| p5 | NPO–Museum | 6 | 7 | 5 (11) | 8 (26) | 6 (14) |
| p6 | NPO–Shelter | 4 | 9 | 7 (8) | 5 (18) | 5 (8) |

*marketing* due to its increasing popularity. Such videos–often seen on social media platforms–are shorter in length [29] with a clear storyline of a few shots. *Headlines* are typically shown in the first few seconds followed by supportive images and a *call to action* (CTA) to encourage user's response. As mobile experiences are fastest growing, a campaign might introduce multiple video versions with various aspect ratios and durations for individual devices and platforms [29, 55].

## Formative Study

To understand how designers create a video prototype, we conducted a formative study and suggest design principles.

### *Study Design*

We recruited six professional designers (including four interaction designers, one UX designer, and one visual designer) from our company via an internal study invitation. Participants reported their familiarity with video creation, web design, and marketing as 4.5, 5, and 4.5 (median) in a 5-point Likert scale. Each participant was assigned to work on one of the static web pages shown in Table 1. We provided a snapshot and the raw assets captured from the page in an editable slide deck using Google Slides. These materials include images (from all foreground or background images, icons, and logos), text (from all text elements excluding the header and the footer), and hexadecimal color codes of all elements.

In each 60-minute session, the task for the participants was to compose a storyboard for a 10-second, 16:9, non-interactive marketing video in the same slide deck given the page assets. To help participants focus on video "storyboarding" that is commonly used in filmmaking [23], we chose not to introduce any advanced video editing tool. The final frames of the storyboard should be ordered and marked with timestamps in seconds. Detailed animated design was not encouraged. After walking through a warm-up task, participants had 35 minutes to complete the task. Although we restricted participants to use only the assets captured from the page, we encouraged them to be creative to rearrange or mismatch the assets, crop images, or edit text. Each designer was compensated with a $25 gift card for their participation.

### *Findings*

All the six participants used the full 35 minutes to complete their storyboards, while four requested extra 3-5 minutes to polish the work. Table 1 presents the number of scenes and assets that designers composed in their creations. On average, each video includes 5 scenes (2 seconds each) and 7.3 shots, where a scene included one to multiple shots by swapping the text or images while maintaining the same visual structure and colors. Designers carefully avoided overloading content. As high as 90% of the scenes are composed by only one major image and one headline (see Figure 2a-c). A long text headline was often broken into two shots for readability (see Figure 2d).

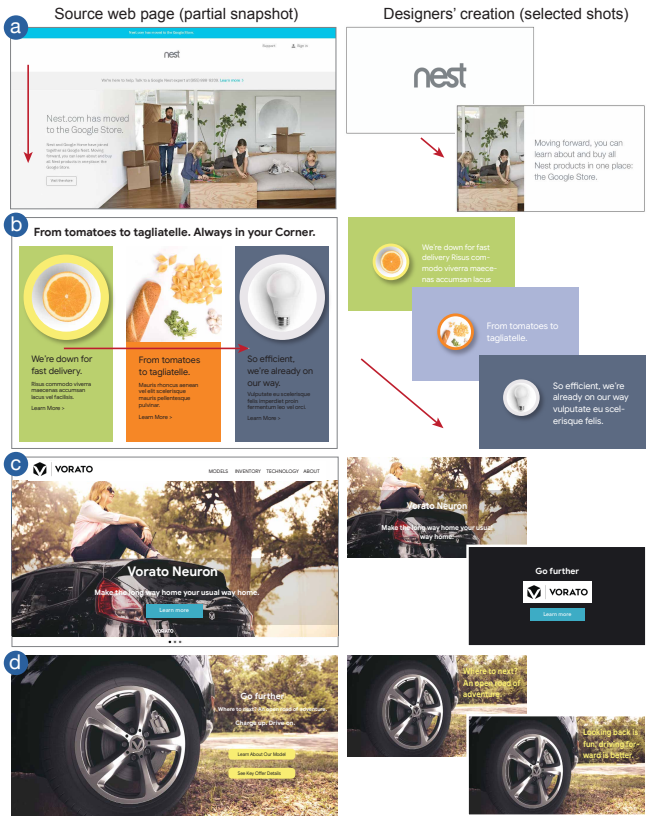


**Figure 2. Video storyboards created by participants in our formative study. Designers often maintain the visual flow of components from the web page to fill a video frame. The source pages were captured in February 2020 from: (a) `https://nest.com/` and (b) to (d) are from two other pages with replaced assets and styles.**

Designers commonly preserved the *visual flow* and graphical compositions from the source web page. For example, Figure 2a shows how the two major sections from the top of the page were rearranged to two ordered scenes. Similarly, the horizontal blocks in Figure 2b are shown as three consecutive scenes. We also noticed that some content was reordered from the flow shown on the page. Designers who iteratively rearranged the frames explained that their strategy was to first visualize the major content and then reason to make adjustment for storytelling.

All the designers carefully examined the *design attributes*. To provide a consistent branding appearance, colors and fonts were picked up from the page to apply to the visual elements, such as background colors and headlines (e.g., Figure 2a and b). Graphical layouts were adjusted to adapt to the frame size.

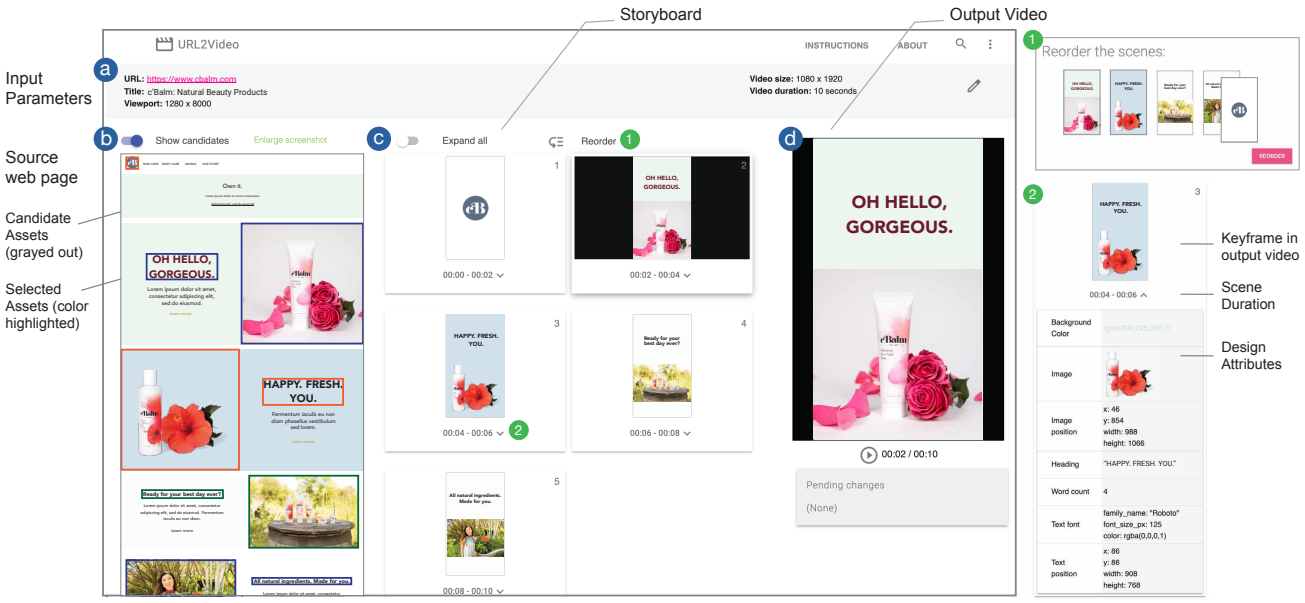


**Figure 3. In URL2Video's user interface, creators specify the input URL to a source page, a viewport (size of the target page view), and the output video parameters (a). URL2Video renders the web page and extracts major visual components, including images and text (b). It composes a series of scenes and visualizes the key frames as a storyboard (c). These components are rendered into an output video that satisfies the input temporal and spatial constraints (d). Users can playback the video, examine the detailed design attributes (c-2), and make adjustments to generate video variation, such as reordering the scenes (c-1). The source web page was captured in February 2020 from a commercial brand, while its assets and styles were replaced.**

Finally, all the designers made the *temporal decisions* at the end of the creation process. All of them walked through the frames back and forth to allocate the timing at least in two or more iterations. Two had to remove frames given the duration constraint. Participants explained that although they considered pacing while composing the graphics, it was challenging to assign the detailed timing for every single frame.

### Design Guidelines

Based on the findings from our formative study and prior work, we propose five design guidelines for creativity tools that convert a web page to a video.

*Provide concise information.* A marketing video promotes a brand, organization, or service in a short attention time span. A tool should generate videos that clearly illustrate the most important messaging and avoid overloading information for viewers to process. Ideally, each major video shot contains only one messaging with simple graphical support.

*Reveal content hierarchy.* Web design inherits marketing intentions to call out the major messaging and information priority [38, 39]. It is important that a tool examines the content ordering on a web page and unfolds the hierarchy of web elements in the output video.

*Transfer design choices.* Visual design attributes of a web page create a brand's look and feel, including colors, images, sizes, fonts, and more. To maintain consistent visual impression, a tool should extract design choices from the source page and apply to video composition when appropriate.

*Suggest temporal allocation.* Making timing decisions for a series of shots can be challenging, especially given a fixed video duration. A creation tool should suggest a video timeline, ideally based on content understanding.

*Review & author.* While a tool makes automatic decision on video creation, it is critical to reveal the detailed attributes of both temporal and visual design and allow creators to iterate via a user interface.

## VIDEO CREATION FROM A WEB PAGE WITH URL2VIDEO

We present URL2Video, a video creation tool that automatically converts content from a web page to a short-form video for digital marketing. We limit our domain to static web pages that contain salient images and headings preserved in an HTML hierarchy. Recent web design trends encourage prominent elements, distinct sections, and layered effect [30]. Our goal is to develop an automatic pipeline that captures the key content from such type of pages and transfers its visual design to the video format. The generated results can support creators exploring video variation for future detailed editing.

We develop an end-to-end solution that renders a web page, extracts its assets and styles, and automatically makes both temporal and visual decisions to compose content in a video given user-specified constraints (see Figure 4). Users can review and modify the design decisions of the video output via URL2Video's user interface to produce more videos (see Figure 3). To help us illustrate the workflow, assume a designer, Maris, who wants to prototype a marketing video from her business web page.

### Web Asset Analysis

To create a video, Maris opens URL2Video's UI and specifies the Uniform Resource Locator (URL) to the source web page of a beauty brand (see Figure 3a left). She chooses a large

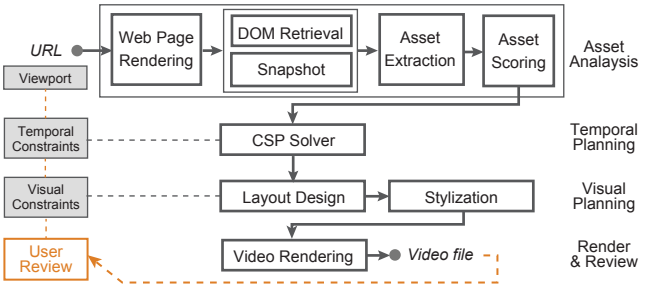


**Figure 4. URL2Video provides an end-to-end solution to generate a video from a web page. Our pipeline contains four stages: asset analysis, video editing with temporal and visual planning, and video rendering & user review. The input parameters such as viewport size and constraints can be modified via URL2Video's user interface (see Figure 3).**

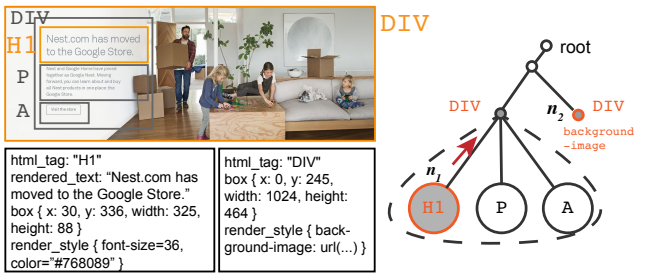


**Figure 5. URL2Video retrieves major nodes from a DOM tree to form a list of asset groups via a bottom-up approach. For example, here an asset group contains a main `H1` heading ($n_1$) on top of a background image (a `DIV` node $n_2$). The example was captured from https://nest.com/.**

viewport (here, 1,280 pixel in width by 8,000 pixel in height) for the system to render the page based on the site's desktop experience with scrollable content. Given these inputs, URL2Video runs a cloud-based rendering engine to retrieve the page's DOM tree and an image snapshot (see Figure 3b).

URL2Video identifies visually-distinguishable elements as a candidate list of *asset groups*. For examples, Maris' page contains a logo (as Asset Group #1) and multiple sections from top to down with salient colors. URL2Video identifies a smaller, text-only segment (as Group #2), followed by a set of sections that each has an `H1` heading and a high-quality product image (as Group #3-#7). In each group, detailed elements including raw text, multimedia files, CSS styles, and locations on the snapshot are captured. URL2Video assigns a priority score to each asset group based on its visual attributes.

### Video Editing and Rendering

In the same UI, Maris specifies the video output constraints, which include temporal (video duration to be 10 seconds) and spatial (aspect ratio to be 9:16) specifications for a short portrait video (see Figure 3a right). Based on the criteria, URL2Video automatically selects and orders the asset groups to optimize the total priority score. URL2Video makes both *temporal* decisions to allocate the materials on a timeline, as well as *visual* decisions to present the assets into individual shots. It then renders the content into a video in the MPEG-4 container format.

### User Authoring

Maris can now playback the output video in the UI (see Figure 3d). She reviews the storyboard that shows an ordered list of thumbnails to each scene (see Figure 3c), which can be expanded to include specific design attributes such as images, text, and color codes of the elements (see Figure 3c-2). Maris finds the video composition reasonable: the video starts with a scene of a logo at the center filled with the background color from Asset Group #1. It follows by a scene with a product image with a headline side-by-side from Group #3, while Group #2 with smaller presence is not included. Now, she wants to make a few changes. In the storyboard, she replaces one scene with a different asset group. She drags the logo scene to the end (see Figure 3c-1). URL2Video generates a new video given these new constraints and updates the new video composition in the interface. Finally, Maris modified the output parameters for a square (1:1) video and can quickly receive a new version with the same asset flow.

## AUTOMATIC VIDEO EDITING

URL2Video automatically makes video edits in both the temporal and spatial domains to present visual assets from a web page. It extracts salient asset groups by DOM retrieval and evaluates their importance. Next, it makes preliminary temporal decisions to allocate assets on a video timeline using constraint-based programming. It then designs graphical presentation by dynamic programming.

### Asset Extraction and Scoring

Given the DOM tree of a source web page, URL2Video retrieves a list of candidate asset groups that contain salient information. Prior research has suggested methods to completely segment a web page from DOM using graph theories [10], context matching [21] or restructuring [39] for applications such as design retargeting. While a fully-segmented page can resolve our needs, we chose a different approach by extracting salient nodes based on HTML annotations (see Figure 5).

We assume that for marketing pages, web designers organize content both visually and semantically so that the information can be traced in a hierarchical HTML structure. Recent web design principles suggest visually segmenting content to help viewers efficiently prioritize information [30]. The information units[1] are distinguishable by visual attributes, such as colors, margins, and sizes. A common group contains a major image that supports a heading placed next to or upon it (see Figure 2a and c). Multiple groups in a grid system could present similar items separated by colors and margin (see Figure 2b).

Define a source DOM tree contains a total of $Z$ nodes, $N_{all} = \{n_z, z \in Z\}$. Our algorithm first looks for major DOM nodes $N_{major} \subseteq N$ that are one of the high-priority HTML tags we selected, including `H1` to `H6` for main headlines, `IMG` or `VIDEO` for multimedia nodes, and CSS attributes that point to multimedia files for background assets. For each $n_j \in N_{major}$, URL2Video creates an asset group $g_i$ with $n_j$ and its container

[1] Since the term "segment" is used both in document and video analysis, here we refer such a visual section in a page as an asset group.

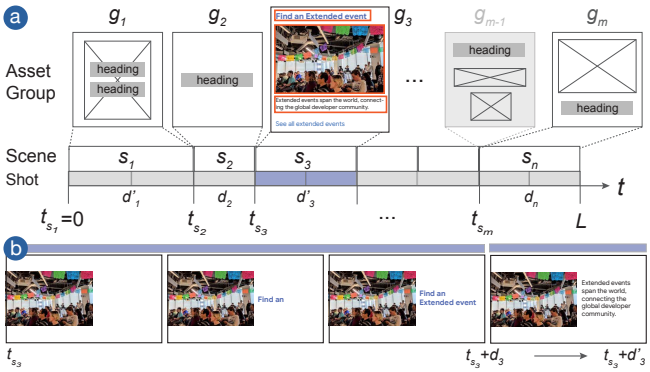


**Figure 6. URL2Video assigns a list of asset groups extracted from a web page to the video timeline based on asset scoring of visual importance (a). For each asset group with a non-zero duration, URL2Video composes a scene with one to more shots and adjusts the timeline for readability (b). Credit: https://events.google.com/io2019/.**

node $n_k$ if its region on the snapshot is above an occupancy threshold. To avoid under-grouping, it merges groups where the contained nodes overlap or are graphically adjacent. This process forms a candidate list of asset groups $G$.

For each asset group $g_i \in G$, URL2Video evaluates the contained nodes and assigns a score $a_i$ that prioritizes by (1) the HTML tags, (2) the ordering from the top of the page, and (3) the region size. In other words, the groups with larger presence shown on the page beginning have the highest scores. Here we do not claim novelty in our asset extraction algorithm, and acknowledge that this method might miss capturing improperly annotated nodes or orphans, which can be manually added to the storyboard in our UI.

### Constraint-Based Temporal Planning

URL2Video organizes assets sequentially to complete the timeline of an output video. We formulate temporal planning of video creation to be an *optimization problem*. Prior work has demonstrated methods to optimize video editing from raw footage [48, 6]. In our domain, given temporal constraints and a list of evaluated assets, URL2Video optimizes the overall score of visual importance. We develop a Constraint-Satisfaction Problem (CSP) solver to determine a preliminary timeline allocation (see Figure 6a).

#### *Variables and Domains*

For each asset group $g_i \in G$, we assign an integer variable, *the duration time* $d$ to present $g_i$ in the output video. Therefore, the variables of our model are $D = \{d_i, i \in m\}$. The domain of $D$ is $[0, L]$, where $L$ is the user-specified video length.

#### *Constraints*

Our solver is bounded by a list of parameterized constraints:

- The sum of duration for $g_i \in G$ is equal to the output video length $L$, i.e., $\sum_{i=1}^{m} d = L$. Groups are presented sequentially without a temporal gap or overlap on the video timeline.
- An asset group $g_i$ can be excluded in the video when $d = 0$.
- To allow viewers to process the content with enough reading time, any selected asset group $g_s \in G_s \subseteq G$ should be visible for a minimum amount of duration: $\forall g_s, 0 < d_{min} \leq d$.


**Figure 7. Assets can be laid out differently given a video's aspect ratio (a). URL2Video combines layouts of images and text elements based on their relative positions on the original page and the output orientation (b). Logo images are placed at the center of a scene, filled with the background color from the source page (c). The example pages were captured in April 2020 with replaced assets.**

- To encourage dynamic content, a video should contain multiple asset groups. Therefore, $\forall g_s, 0 < d \leq d_{max}$, where $d_{max}$ is determined by $L$ and number of groups $m$.
- Any asset group that users explicitly specifies via our UI will be included, i.e., $\forall g_u \in G_{user}, d > 0$.

#### *Objective Function*

Finally, our CSP solver finds a solution that maximizes the total score of $g_i \in G$ while satisfying the temporal constraints:

$$\max \sum_{i=1}^{m} a_i \times d_i$$

In this way, our optimization suggests a sequence of asset groups that have the highest priority score, where each selected group $g_s \in G_s$ has a duration $d_s$ mapped to the video timeline of the total video length $L$. This time allocation can later be adjusted based on visual planning.

### Visual Planning

For each selected asset group $g_s$ of an assigned duration $d$, URL2Video plans a graphical design to present its text, image, or video assets in a video output. There are two goals of visual planning from our proposed design guidelines: (1) each video shot provides concise information, and (2) the visual design is consistent to the source page. We treat this as a dynamic programming problem to generate a series of video scenes, similar to prior work on document generation [36, 53].

**Content Selection:** To achieve the first goal, we restrict each shot to contain at most three dominant elements, including one major multimedia asset (an image or a video), one non-video asset (a headline or another image), and a background color

layer. URL2Video selects an asset subset $g'_s$ from $g_s$ for each scene of time $d_s$.

**Graphical Layout:** Next, URL2Video decides the graphical presentation for the assets in $g'_s$. Learned from designers' work in our formative study, we define three graphical layouts for multimedia assets and text respectively (see Figure 7a). URL2Video considers the original design from the source page. For example, Figure 7b presents three combinations of an image-text pair, including side by side, a full image with a full heading, and a full image with the heading on one side. We compute the ratio of asset placement while considering the aspect ratio of the video output. Additional layout templates can be included to support more input assets as prior work shows [36, 37], although it's beyond the scope of this paper.

**Stylization:** URL2Video applies the style of a text asset from the source page, including the font choice, size, weight, and horizontal alignment (see Figure 8). To enable text legibility, we cross-check and replace the text color contrast with its background color if not placed on an image based on the accessibility guidelines [34, 11]. We also define a minimum and maximum font sizes for readability and visual balance.

**Timeline Adjustment:** While URL2Video's default strategy is to include a diverse set of asset groups in a video, we design another strategy for longer videos that allow timeline adjustment. For a remained asset in $g'_s$, such as a second heading in Figure 6b, URL2Video replaces one of the shown assets and extends the duration from $d$ to $d'$ to enable viewers to review more information. URL2Video then shortens or removes one remaining asset group with the lowest score from $G_s$ to maintain the total video duration $L$.

**Animation:** Adjusting the presentation timing of assets can make a video more dynamic and engaging [66]. We apply animation at two levels (see Figure 6b). First, for an multimedia-text pair, we delay the second asset from the visual flow by 0.5 seconds. An asset is shown first if positioned on the left in a landscape video or on the top in a portrait video. Second, we apply word-by-word animation for a short text title, or words-by-words for a longer title. To avoid over-animating, we do not apply effects for the extended asset in a scene (such as the replaced text heading shown in Figure 6b). While prior work has suggested applying motion to static content for engagement [61, 9], we chose not to manipulate visual assets to avoid misinterpretation.

## IMPLEMENTATION

We developed the end-to-end pipeline in C++ upon several existing tools. First, the web page analysis tool uses an internal library developed by our organization. It is similar to available HTML parsers and web renderers for DOM tree and snapshot retrieval. Second, the CSP solver uses libraries from Google's OR-Tools [33]. Finally, we built our video rendering system on MediaPipe [32], OpenCV [7], and FFMpeg [18] that are all open-sourced. Our renderer composes a series of timestamped asset layers into video frames and encodes into a MP4 video. Text shaping and rendering are based on open-sourced libraries HarfBuzz [19] and Skia [20]. In our current implementation, font choices from a web page might not always be available if beyond a constrained set of font families we supported. Our approach is to replace an unsupported font with a default option, Roboto. We leave automatic font replacement [65, 47] to the future work.

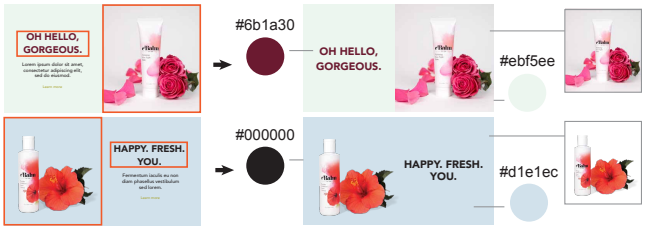


**Figure 8. URL2Video converts the design styles from the DOM nodes in a web page to the asset layers in a video, including colors, font styles, text alignment, and ratio.**

**Table 2. Analysis of URL2Video's captured assets and output video composition for 50 pages of 9 categories.**

| | Page | # of source units | | | # of output layers | | | Composition | |
|---|---|---|---|---|---|---|---|---|---|
| | Count | Image | Video | Text | Image | Video | Text | Scene | Shot |
| Cars | 2 | 8.5 | 0.0 | 92.0 | 4.0 | 0.0 | 10.5 | 5.0 | 17.0 |
| Electronics | 6 | 12.3 | 0.0 | 104.2 | 3.2 | 0.0 | 6.0 | 5.0 | 13.0 |
| Fashion | 4 | 13.8 | 0.0 | 91.3 | 4.0 | 0.0 | 5.5 | 5.0 | 14.5 |
| Food & Drink | 13 | 17.8 | 0.1 | 115.6 | 5.1 | 0.1 | 7.8 | 4.8 | 14.5 |
| Grocery | 4 | 20.0 | 0.0 | 64.8 | 4.8 | 0.0 | 3.0 | 5.0 | 12.8 |
| NPO-Education | 7 | 16.6 | 0.0 | 110.3 | 3.9 | 0.0 | 12.4 | 5.0 | 20.3 |
| NPO-Health | 3 | 17.7 | 0.0 | 121.7 | 4.7 | 0.0 | 16.0 | 5.0 | 25.0 |
| NPO-Animal | 2 | 16.5 | 0.0 | 76.5 | 5.0 | 0.0 | 16.0 | 5.0 | 22.5 |
| Service | 9 | 18.2 | 0.7 | 72.0 | 3.7 | 0.3 | 5.4 | 4.9 | 13.0 |
| *Average* | 5.6 | 15.7 | 0.1 | 94.2 | 4.2 | 0.0 | 9.2 | 5.0 | 17.0 |
| *Total* | 50 | 823 | 7 | 4874 | 212 | 4 | 408 | 246 | 789 |

## RESULTS

To examine the generality of URL2Video and the quality of the generated videos, we created a dataset of 50 existing web pages and describe the video output examples.

*Dataset.* We selected 50 web pages in 9 categories based on the following criteria: A page (1) illustrates a brand or an organization, (2) is mostly static without user input or complicated animation, and (3) contains a visual structure composed with text and images. Embedded videos clips annotated by the HTML tag `VIDEO` can be included.

*Methods.* We performed the end-to-end pipeline using URL2Video and were able to create videos for all the 50 web pages. There are three source pages that produced text-only videos as our engine was not able to retrieve the multimedia assets due to access restriction. 32 pages required author intervention to retrieve at least one embedded image of Scalable Vector Graphics (SVG) or other formats. We examined the annotations of each page and corrected five asset groups on average, including adding unselected nodes (e.g., a missing `DIV` of a logo) and merging nodes to the same group. This correction process took 3-5 minutes per page via a review UI. Then, given an annotated page, it required 54 seconds on average to perform asset scoring, planning, and rendering to generate a video output. There were 10 additional pages out of the 50 pages that we excluded as our web rendering engine received no response from DOM retrieval.

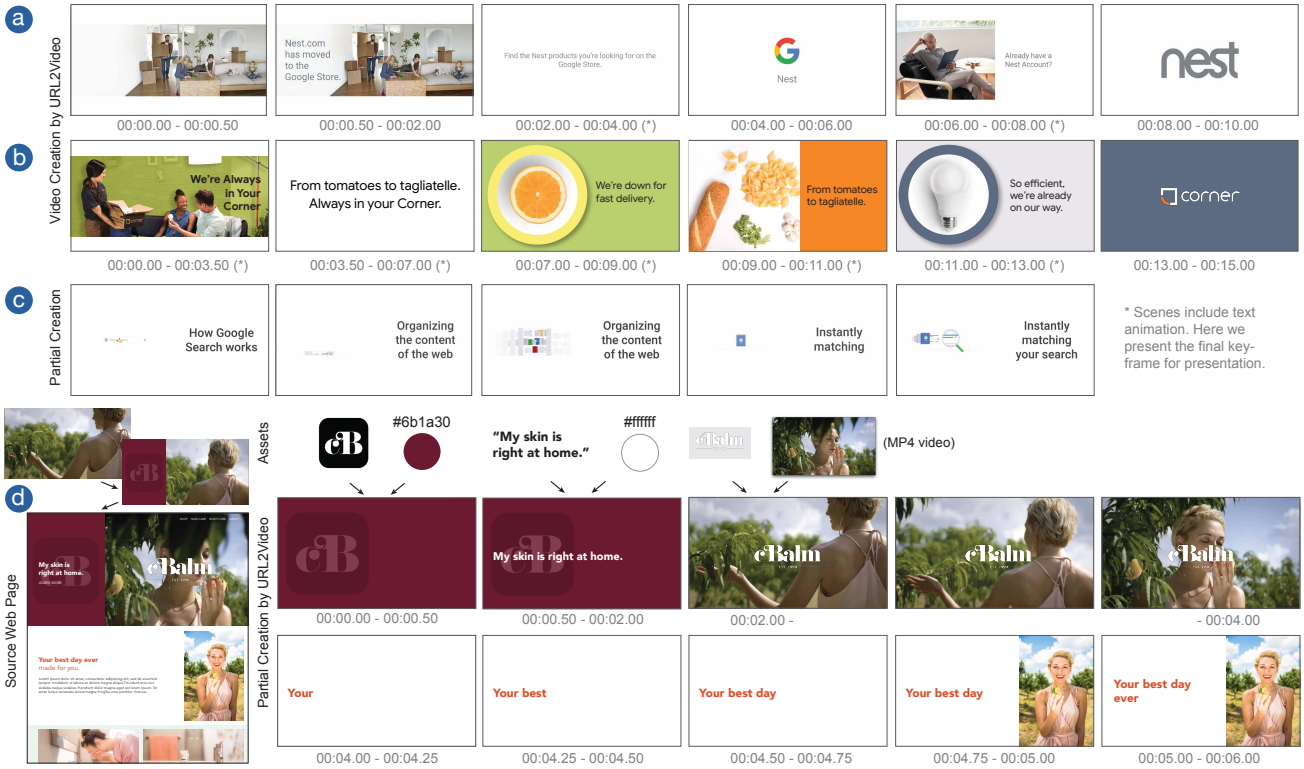


**Figure 9. Example automatically-generated results by URL2Video. (a) and (b) are outputs of the pages shown in Figure 2 from https://nest.com/ and a page with replaced assets. (c) shows a partial video output for a page that embeds video clips from https://google.com/search/howsearchworks/. (d) shows how URL2Video composes for a page with more dynamic content.**

*Results.* We performed our pipeline to create videos of three aspect ratios (16:9, 9:16, and 1:1) and 10 and 15 seconds (i.e., 6 output variations per source web page). Table 2 shows an analysis of the assets URL2Video captured and the number of asset layers used in 10-second 16:9 output videos with our dataset. In total, URL2Video extracted 823 images, 7 videos, and 4874 text elements from 50 pages. For a 10-second output video, on average it creates 4.92 scenes composed by a total of 15.78 shots.

Figure 9 shows a sample set of videos created by URL2Video. Overall, the automatically-generated videos followed the design guidelines from our formative study: URL2Video presents concise information in a similar visual flow from the source pages. It transfers the design attributes to a video and applies minimal animation for engagement. Below we highlight some results according to their presentation.

**Visual Composition:** We observed that the most common types of graphical composition on a marketing web page to a video includes: (1) a background image fills a section with a heading on top of it (e.g., Figure 9a-2 and b-1), (2) a heading that fills a section (e.g., Figure 9a-3 and b-2), and (3) an image accompanied by a heading around it (e.g., Figure 9a-5 and b-3 to 5). URL2Video was able to preserve their visual structure and color decisions. However, we acknowledge that some layouts could be optimized to fill more spaces in a frame.

**Temporal Composition:** URL2Video correctly organizes the scenes based on the visual flow from the source pages and applies a small degree of animation. Take Figure 9d for example. A heading and an image are presented sequentially to guide viewers' attention. Although we do not yet handle scene-by-scene transition, such micro animation provides dynamics to a video beyond a slideshow look-and-feel.

**Support of Multimedia:** URL2Video supports pages with video assets, which are increasing popular on modern websites. In Figure 9c, URL2Video places video clips along with a heading. In Figure 9d, a video is on the background layer with a static image on the top. We noticed that such a combination makes a video more dynamic.

**Language and Style Variation:** While the majority of our dataset is in English, we tested how URL2Video supports different languages and styles with a multinational chain. We noticed that the chain maintains a similar visual structure of their home pages from 10 countries. Our pipeline was able to create videos for each page presenting various promotions, localized assets, and languages. However, there were pages from this chain with multiple layers, grids, and animation that URL2Video fails to fully capture.

## USER EVALUATION

We evaluated the generated videos by URL2Video with 8 professional designers to understand their perception of video quality and the usability of our user interface. In addition, we conducted an online survey with 65 participants to understand how general audience perceives the video quality.

### Study I: Designer Inspection

We compared URL2Video with a *screenshot-based* Baseline for the six pages (denoted p1-p6) in Table 1. We invited the designers from our formative study for a 30-minute remote session to review the videos from the pages that they previously worked on. Five were able to join the study (denoted D1-D5) and received a $15 gift card for their participation. To collect feedback from designers who were *unfamiliar* with the source pages, we invited three additional designers (denoted D6-D8) from the same candidate pool for a 60-minute remote session. Each reviewed videos from two distinct pages and received a $25 gift card for participation. In this way, videos from each page (except for p6) was reviewed by two designers, including one with content familiarity and one without.

*Materials.* For each of the six pages, we created two videos for the Baseline condition. We manually captured screenshots of the logo and the first four major sections to compose a 10-second video in 16:9 and 9:16 aspect ratios. Each screenshot was shown two seconds regardless the information density. We created 4 videos with URL2Video, including 10-second videos in 16:9, 9:16, and 1:1 aspect ratios and a 15-second 16:9 video for comparison. Therefore, we created 12 videos for Baseline and 24 videos for URL2Video to review.

*Procedure.* Each session includes video evaluation, UI operation, and a questionnaire. We randomized the videos of two conditions and played each video once, followed by 5-point Likert-scale questions as follows:

**(Q1)** It is easy to follow this video.
**(Q2)** The video communicates important information.
**(Q3)** I understand the message in this video.
**(Q4)** The video appears professionally designed.
**(Q5)** It is pleasing to watch this video.
**(Q6)** The pace of the video is about right.

The scale is ranged from Strongly Disagree (1) to Strongly Agree (5) for **Q1** to **Q5**, and from Too Slow (1) to Too Fast (5) for **Q6**. The three designers new to materials did not review any pages before this evaluation. After evaluating all the videos, we introduced our UI showing the 16:9 10-second video created by URL2Video. Designers walked through the functionality, inspected design details of the video, and suggested edits. Finally, designers completed a questionnaire.

*Results.* Designers found the concept of converting design from a web page to videos straightforward (Median=4), and agreed that it was faster to create a video prototype with URL2Video over without its support (M=4.5 from D1-5 who participated in the formative study). Designers commented on URL2Video's advantages as, "*Quickly extract assets for further customization; Easy information hierarchy from website that can be translated into a good visual narrative*" (D3), "*Produce quick marketing videos for small businesses & people who have no video editing experience*" (D5), and "*Able to take a first pass, which can save a lot of time!*" (D7). Below we aggregated the results from all videos of the two conditions[2].

[2]The median rating performed the same if we compared videos of all aspect ratios and duration with only videos in 16:9 and 9:16 as the Baseline condition.

**URL2Video outperformed Baseline.** All designers found URL2Video easy to follow (M=4) compared to the Baseline (M=2) to **Q1**. They consistently pointed out that web pages commonly contain a great amount of information, and the content is not meant to be processed within a short amount of time. Therefore, screenshots would never be ideal for video viewing. Similarly, they understood the message from URL2Video (M=4) compared to the Baseline (M=2) to **Q3**, and the videos communicated important information (M=4) over the Baseline (M=3) to **Q2**. Designers explained, "*it's easy to read the headlines and reason about the message (with URL2Video)*" (D4) and "*the extracted content makes a video easier to follow, but the storyline could be improved*" (D6).

**URL2Video paced adequately.** Designers rated Baseline videos too fast (M=5), while the pace of URL2Video was neutral (M=3, neither too fast nor too slow) to **Q6**. This could be due to the information load while the Baseline required more time to process the content in each shot.

**Require design iteration.** However, designers were not satisfied with the design quality (**Q4**) in both conditions (M=2) and were neutral to **Q5** in terms of pleasure (compared to the Baseline M=2). They commonly noticed the inappropriate typefaces, text overlays, underoptimized layouts, and extraneous margin from the our output videos. D7 immediately commented on the font choice of videos for p2 (a fashion web page with a special font that URL2Video did not support), even if she had not seen the source design. Some designers suggested improving the text animation.

Designers found the storyboard with design details useful in our UI (M=4), but they looked for more editing capability to iterate on the computationally-generated design. Designers commented on the shortcomings of URL2Video as, "*Not yet at a level where I would trust it without some design/human oversight*" (D7) and "*The video quality will be heavily relied on the quality of the source webpage*" (D2). We acknowledged the limitation of quality in design although our results achieved a reasonable level of conveying a brand's message and outperformed a naive Baseline.

### Study II: Audience Survey

To understand how general audience perceives the video quality, we conducted an online survey to compare URL2Video with *designers' creations*.

*Materials.* For each page, we generated a 10-second 16:9 video from the formative study as the Baseline condition (denoted as B1 to B6). We acknowledged that designers' creation from the slide deck was not intended to be a final marketing video. However, we aimed to understand if the audience accepted a video prototype created by experts within limited time in 35 minutes over computationally-generated outputs.

We updated URL2Video's algorithms based on designers' feedback from Study I to generate new 10-second 16:9 videos as the Control condition (denoted as C1 to C6). The changes include: placing the logos at the final scene, increasing font sizes, and fixing text overlay. However, we did not replace the missing fonts or manipulate the assets. We included two additional videos with embedded video clips (denoted as C7

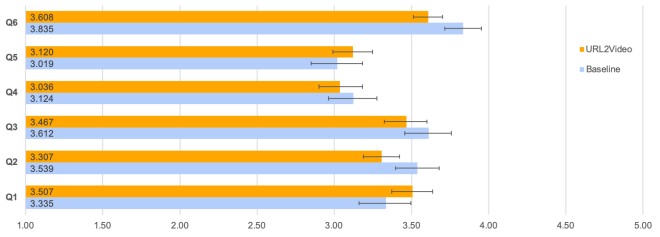


**Figure 10. In our online survey with 65 participants, we observed no statistically significant difference between professional designers' creations and URL2Video videos.**

and C8; see Figure 9c and d). The goal was to investigate whether including video assets led to better quality.

*Method.* We distributed our online survey via multiple internal listservs for a wide range of audience that includes professionals of all roles in our organization. Each survey included 7 videos, three from B1-B6 (Baseline), three from C1-C6 (URL2Video) for pair-wise comparison, and one video from C7 or C8 (video-embedded URL2Video). The videos were randomized ordered and played only once for the same list of six Likert-scale questions as used in Study I. Participants did not receive any monetary incentive for survey completion.

*Result.* We received 65 unique responses to the online survey. Each of the 14 videos was reviewed by at least 25 participants (due to randomization). We analyzed the data and reported results based on bootstrap method [52] (see Figure 10). Survey responses from participants indicated no statistically significant difference between the Baseline and URL2Video videos for most of the questions: easy to follow (**Q1**), communicating important information (**Q2**), understanding the message (**Q3**), appearing professionally designed (**Q4**), pleasing to watch the video (**Q5**). The pacing of the Baseline was reported to be statistically significantly faster than URL2Video (**Q6**), while both conditions were considered to be faster than ideal.

The fact that participants reported no significant difference in professionality (**Q4**) between the two conditions is surprising: the Baseline condition was hand-created by professional designers. While the designers worked within the constraints of video prototyping, we consider this an indication that URL2Video is able to perform sufficiently professionally for our users. Participants understood the message no better from the Baseline (**Q3**) and reported that the videos communicated the same important information (**Q2**). In Study I, professional designers identified that URL2Video might miss important information that wasn't in a heading on the web page, but surveyed viewers of the completed video did not make such a distinction. Finally, C8, which embedded video clips, received significantly higher ratings on design quality (**Q4**) and easiness to follow (**Q1**).

We received positive comments from respondents: "*It was fun to realize that I couldn't make good guesses about which were generated by models and which were not*", "*Videos are automatically generated? Very impressive!*", and "*I'd be surprised if the landing page from any video looks visually similar. Usually only top brands have enough budget for this.*"

## DISCUSSION AND OPPORTUNITIES

Overall, we received positive feedback from designers and general audience on their perception in video quality of our computationally-generated results. Designers found URL2Video useful to extract important assets for an initial pass of video composition. Audience found the video quality similar to designers' prototypes. Below we describe more of URL2Video's limitations and opportunities.

*Source page design.* As designers from Study I commented, the video quality heavily depends on the source web page. URL2Video does not handle pages that have limited quality assets, lack of a hierarchy, require user input, or are over-complicated with intensive animations or layers. Such pages will result in a video with sparse or no content. We suggest providing an interface for users to guide or modify the automatic editing decisions for iterative creation [1]. Moreover, URL2Video does not optimize the layouts for portrait or square videos and would often result in significant spaces (such as Figure 7b). To better support mobile experiences, future work could learn adaptive design from source pages optimized by web designers [8].

*Multiple compositions and audiences.* A web page may contain several topics for audience's needs. Our current technique that places information from the page top into a video might not be suitable for all designs. Future research should include better content understanding to support more advanced asset selection, storytelling, and CTA suggestion. While this paper focuses on marketing videos, the approach can possibly apply to other domains, including instructional and announcement videos for learning web content via the animated format.

*Animation and audio.* Our current pipeline supports well-structured, concise pages given the assumption that complicated content often leads to less aesthetics [43, 35]. While we provide limited animation in a video, our designer participants also addressed the importance of transitions and background music to enhance video quality. We are experimenting applying the assets to animated templates, and synthesizing voiceover from text assets to support the visuals. This might further support text-driven pages by better document understanding as recent research has demonstrated [31, 41].

## CONCLUSION

This paper introduced URL2Video, an automatic approach that converts a web page into a short video. URL2Video captures quality materials and design styles extracted from a source page. Given a set of user-specified temporal and visual constraints, URL2Video's design engine organizes the assets into a sequence of shots and renders to a video output. Our user interface presents the selected assets from the web page and the video composition. Creators can review the automatic design decisions, modify constraints, a few parameters, and refine the video. We evaluated URL2Video's automatically-generated results from a dataset we created and compared with designers' creations through interviews and an online survey. The evaluation suggests that URL2Video effectively composes videos from a web page and could support designers by bootstrapping the video creation process.

## ACKNOWLEDGEMENT

We greatly thank all the designers participated in our studies for their creation and 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): Jordan Canedy, Brian Curless, Nathan Frey, Madison Le, Alireza Mahdian, Justin Parra, Emily Ryan, Mogan Shieh, Sandor Szego, and Weilong Yang.

## REFERENCES

[1] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul Bennett, Kori Inkpen, Jaime Teevan, Ruth Kikin-Gil, and Eric Horvitz. 2019. Guidelines for Human-AI Interaction. In *CHI 2019*. ACM. https://www.microsoft.com/en-us/research/publication/guidelines-for-human-ai-interaction/ Best Paper Honorable Mention.

[2] Daniel Arijon. 1991. *Grammar of the film language*. Silman-James Press.

[3] Edward O. Benson and David R. Karger. 2013. Cascading Tree Sheets and Recombinant HTML: Better Encapsulation and Retargeting of Web Content. In *Proceedings of the 22Nd International Conference on World Wide Web (WWW '13)*. ACM, New York, NY, USA, 107–118. DOI: http://dx.doi.org/10.1145/2488388.2488399

[4] S. Berrani, H. Boukadida, and P. Gros. 2013. Constraint Satisfaction Programming for Video Summarization. In *2013 IEEE International Symposium on Multimedia*. 195–202.

[5] 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. DOI: http://dx.doi.org/10.1145/2185520.2185563

[6] H. Boukadida, S. Berrani, and P. Gros. 2017. Automatically Creating Adaptive Video Summaries Using Constraint Satisfaction Programming: Application to Sport Content. *IEEE Transactions on Circuits and Systems for Video Technology* 27, 4 (2017), 920–934.

[7] G. Bradski. 2000. The OpenCV Library. *Dr. Dobb's Journal of Software Tools* (2000).

[8] Peter Brusilovsky and Mark T. Maybury. 2002. From Adaptive Hypermedia to the Adaptive Web. *Commun. ACM* 45, 5 (May 2002), 30–33. DOI: http://dx.doi.org/10.1145/506218.506239

[9] Y. Cao, X. Pang, A. B. Chan, and R. W. H. Lau. 2017. Dynamic Manga: Animating Still Manga via Camera Movement. *IEEE Transactions on Multimedia* 19, 1 (Jan 2017), 160–172. DOI: http://dx.doi.org/10.1109/TMM.2016.2609415

[10] Deepayan Chakrabarti, Ravi Kumar, and Kunal Punera. 2008. A Graph-theoretic Approach to Webpage Segmentation. In *Proceedings of the 17th International Conference on World Wide Web (WWW '08)*. ACM, New York, NY, USA, 377–386. DOI: http://dx.doi.org/10.1145/1367497.1367549

[11] Huiwen Chang, Ohad Fried, Yiming Liu, Stephen DiVerdi, and Adam Finkelstein. 2015. Palette-based Photo Recoloring. *Transactions on Graphics (Proceedings of SIGGRAPH)* (2015).

[12] 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. DOI: http://dx.doi.org/10.1145/1719970.1720014

[13] Jun-Cheng Chen, Wei-Ta Chu, Jin-Hau Kuo, Chung-Yi Weng, and Ja-Ling Wu. 2006. Tiling Slideshow. In *Proceedings of the 14th ACM International Conference on Multimedia (MM '06)*. Association for Computing Machinery, New York, NY, USA, 25–34. DOI: http://dx.doi.org/10.1145/1180639.1180653

[14] Pei-Yu Chi, Joyce Liu, Jason Linder, Mira Dontcheva, Wilmot Li, and Bjoern 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. DOI: http://dx.doi.org/10.1145/2501988.2502052

[15] Lydia B. Chilton, Savvas Petridis, and Maneesh Agrawala. 2019. VisiBlends: A Flexible Workflow for Visual Blends. 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 172, 14 pages. DOI: http://dx.doi.org/10.1145/3290605.3300402

[16] Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. 2017. Rico: A Mobile App Dataset for Building Data-Driven Design Applications. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. ACM, New York, NY, USA, 845–854. DOI: http://dx.doi.org/10.1145/3126594.3126651

[17] Biplab Deka, Zifeng Huang, and Ranjitha Kumar. 2016. ERICA: Interaction Mining Mobile Apps. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 767–776. DOI: http://dx.doi.org/10.1145/2984511.2984581

[18] FFMpeg Developers. 2020a. FFMpeg tool. (2020). Retrieved March, 2020 from http://ffmpeg.org/

[19] HarfBuzz Developers. 2020b. HarfBuzz text shaping engine. (2020). Retrieved March, 2020 from http://harfbuzz.org/

[20] Skia Developers. 2020c. Skia Graphics Library. (2020). Retrieved March, 2020 from https://skia.org/

[21] David Fernandes, Edleno Silva de Moura, Altigran Soares da Silva, Berthier Ribeiro-Neto, and Edisson Braga. 2011. A Site Oriented Method for Segmenting Web Pages. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*. ACM, New York, NY, USA, 215–224. DOI: http://dx.doi.org/10.1145/2009916.2009949

[22] 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. DOI: http://dx.doi.org/10.1145/3306346.3323028

[23] Dan B Goldman, Brian Curless, David Salesin, and Steven M. Seitz. 2006. Schematic Storyboarding for Video Visualization and Editing. In *ACM SIGGRAPH 2006 Papers (SIGGRAPH '06)*. Association for Computing Machinery, New York, NY, USA, 862–871. DOI:http://dx.doi.org/10.1145/1179352.1141967

[24] Tom Horak, Andreas Mathisen, Clemens N. Klokmose, Raimund Dachselt, and Niklas Elmqvist. 2019. Vistribute: Distributing Interactive Visualizations in Dynamic Multi-Device Setups. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM, New York, NY, USA, Article 616, 13 pages. DOI: http://dx.doi.org/10.1145/3290605.3300846

[25] 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. DOI: http://dx.doi.org/10.1145/3290605.3300311

[26] Z. Hussain, M. Zhang, X. Zhang, K. Ye, C. Thomas, Z. Agha, N. Ong, and A. Kovashka. 2017. Automatic Understanding of Image and Video Advertisements. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1100–1110. DOI: http://dx.doi.org/10.1109/CVPR.2017.123

[27] Zaeem Hussain, Mingda Zhang, Xiaozhong Zhang, Keren Ye, Christopher Thomas, Zuha Agha, Nathan Ong, and Adriana Kovashka. 2017. Automatic Understanding of Image and Video Advertisements. 1100–1110. DOI: http://dx.doi.org/10.1109/CVPR.2017.123

[28] The Interactive Advertising Bureau (IAB). 2012. Responsive Design and Ad Creative: An IAB Report. (2012). Retrieved April, 2020 from https://www.iab.com/insights/responsive-design-and-ad-creative-an-iab-report/

[29] The Interactive Advertising Bureau (IAB). 2019. Video Advertising Spend Report 2019. (2019). Retrieved April, 2020 from https://www.iab.com/insights/ad-spend-report-2019/

[30] Google Inc. 2019. Accessibility - Material Design. (2019). Retrieved August, 2019 from https://material.io/design/usability/accessibility.html

[31] Google Inc. 2020a. Document AI. (2020). Retrieved April, 2020 from https://cloud.google.com/solutions/document-ai

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

[33] Google Inc. 2020. OR-Tools. (2020). Retrieved April, 2020 from https://developers.google.com/optimization

[34] Google Inc. 2020. Text legibility. (2020). Retrieved April, 2020 from https://material.io/design/color/text-legibility.html

[35] Melody Y. Ivory, Rashmi R. Sinha, and Marti A. Hearst. 2001. Empirically Validated Web Page Design Metrics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '01)*. Association for Computing Machinery, New York, NY, USA, 53–60. DOI:http://dx.doi.org/10.1145/365024.365035

[36] Charles Jacobs, Wilmot Li, Evan Schrier, David Bargeron, and David Salesin. 2003. Adaptive Grid-Based Document Layout. In *ACM SIGGRAPH 2003 Papers (SIGGRAPH '03)*. Association for Computing Machinery, New York, NY, USA, 838–847. DOI:http://dx.doi.org/10.1145/1201775.882353

[37] 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). DOI: http://dx.doi.org/10.1007/s11042-016-4275-4

[38] Ranjitha Kumar, Arvind Satyanarayan, Cesar Torres, Maxine Lim, Salman Ahmad, Scott R. Klemmer, and Jerry O. Talton. 2013. Webzeitgeist: Design Mining the Web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 3083–3092. DOI: http://dx.doi.org/10.1145/2470654.2466420

[39] Ranjitha Kumar, Jerry O. Talton, Salman Ahmad, and Scott R. Klemmer. 2011. Bricolage: Example-based Retargeting for Web Design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 2197–2206. DOI: http://dx.doi.org/10.1145/1978942.1979262

[40] 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. DOI: http://dx.doi.org/10.1145/3072959.3073653

[41] 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. DOI: http://dx.doi.org/10.1145/3313831.3376519

[42] Thomas F. Liu, Mark Craft, Jason Situ, Ersin Yumer, Radomir Mech, and Ranjitha Kumar. 2018. Learning Design Semantics for Mobile Apps. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST '18)*. ACM, New York, NY, USA, 569–579. DOI: http://dx.doi.org/10.1145/3242587.3242650

[43] Aliaksei Miniukovich and Maurizio Marchese. 2020. Relationship Between Visual Complexity and Aesthetics of Webpages. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. DOI: http://dx.doi.org/10.1145/3313831.3376602

[44] Makoto Nakajima, Daisuke Sakamoto, and Takeo Igarashi. 2014. Offline Painted Media for Digital Animation Authoring. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. Association for Computing Machinery, New York, NY, USA, 321–330. DOI: http://dx.doi.org/10.1145/2556288.2557062

[45] Michael Nebeling and Anind K. Dey. 2016. XDBrowser: User-Defined Cross-Device Web Page Designs. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 5494–5505. DOI: http://dx.doi.org/10.1145/2858036.2858048

[46] Peter ODonovan, Aseem Agarwala, and Aaron Hertzmann. 2014. Learning Layouts for Single-PageGraphic Designs. *IEEE Transactions on Visualization and Computer Graphics* 20, 8 (Aug. 2014), 1200–1213. DOI: http://dx.doi.org/10.1109/TVCG.2014.48

[47] Peter O'Donovan, Jundefinednis Lundefinedbeks, Aseem Agarwala, and Aaron Hertzmann. 2014. Exploratory Font Selection Using Crowdsourced Attributes. *ACM Trans. Graph.* 33, 4, Article Article 92 (July 2014), 9 pages. DOI: http://dx.doi.org/10.1145/2601097.2601110

[48] R. Ogata, Yuichi Nakamura, and Y. Ohta. 2004. Computational video editing model based on optimization with constraint-satisfaction. 688 – 693 vol.2. DOI: http://dx.doi.org/10.1109/ICICS.2003.1292544

[49] X. Pang, Y. Cao, R. Lau, and A. B. Chan. 2016a. Look Over Here: Attention-Directing Composition of Manga Elements. *ACM Transactions on Graphics (Proc. of SIGGRAPH Asia 2016)* 36 (2016). Issue 5.

[50] Xufang Pang, Ying Cao, Rynson W. H. Lau, and Antoni B. Chan. 2016b. Directing User Attention via Visual Flow on Web Designs. *ACM Trans. Graph.* 35, 6, Article Article 240 (Nov. 2016), 11 pages. DOI: http://dx.doi.org/10.1145/2980179.2982422

[51] Phuong Pham and Jingtao Wang. 2019. AttentiveVideo: A Multimodal Approach to Quantify Emotional Responses to Mobile Advertisements. *ACM Trans. Interact. Intell. Syst.* 9, 2–3, Article Article 8 (March 2019), 30 pages. DOI: http://dx.doi.org/10.1145/3232233

[52] 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. DOI: http://dx.doi.org/10.1145/1066157.1066224

[53] Evan Schrier, Mira Dontcheva, Charles Jacobs, Geraldine Wade, and David Salesin. 2008. Adaptive Layout for Dynamically Aggregated Documents. In *Proceedings of the 13th International Conference on Intelligent User Interfaces (IUI '08)*. Association for Computing Machinery, New York, NY, USA, 99–108. DOI:http://dx.doi.org/10.1145/1378773.1378787

[54] Chengyao Shen and Qi Zhao. 2014. Webpage Saliency. In *ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (Eds.). Springer International Publishing, Cham, 33–46.

[55] Kristen Shipley. 2019. Why brands are repurposing longer video ads into bumper ads. (2019). Retrieved April, 2020 from https://www.thinkwithgoogle.com/advertising-channels/bumper-video-ads/

[56] Qingkun Su, Xue Bai, Hongbo Fu, Chiew-Lan Tai, and Jue Wang. 2018. Live Sketch: Video-Driven Dynamic Deformation of Static Drawings. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–12. DOI: http://dx.doi.org/10.1145/3173574.3174236

[57] 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. DOI: http://dx.doi.org/10.1145/2984511.2984569

[58] 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. DOI:http://dx.doi.org/10.1145/3173574.3173890

[59] Nora S. Willett, Rubaiat Habib Kazi, Michael Chen, George Fitzmaurice, Adam Finkelstein, and Tovi Grossman. 2018. A Mixed-Initiative Interface for Animating Static Pictures. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology (UIST '18)*. Association for Computing Machinery, New York, NY, USA, 649–661. DOI: http://dx.doi.org/10.1145/3242587.3242612

[60] Nora S. Willett, Wilmot Li, Jovan Popovic, Floraine Berthouzoz, and Adam Finkelstein. 2017. Secondary Motion for Performed 2D 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, 97–108. DOI:http://dx.doi.org/10.1145/3126594.3126641

[61] Xuemiao Xu, Liang Wan, Xiaopei Liu, Tien-Tsin Wong, Liansheng Wang, and Chi-Sing Leung. 2008. Animating Animal Motion from Still. *ACM Trans. Graph.* 27, 5, Article Article 117 (Dec. 2008), 8 pages. DOI: http://dx.doi.org/10.1145/1409060.1409070

[62] Keren Ye, Kyle Buettner, and Adriana Kovashka. 2018. Story Understanding in Video Advertisements. (2018).

[63] Wen-tau Yih, Joshua Goodman, and Vitor R. Carvalho. 2006. Finding Advertising Keywords on Web Pages. In *Proceedings of the 15th International Conference on World Wide Web (WWW '06)*. Association for Computing Machinery, New York, NY, USA, 213–222. DOI:http://dx.doi.org/10.1145/1135777.1135813

[64] Arianna Yuan and Yang Li. 2020. Modeling Human Visual Search Performance on Realistic Webpages Using Analytical and Deep Learning Methods. 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. DOI:http://dx.doi.org/10.1145/3313831.3376870

[65] Nanxuan Zhao, Ying Cao, and Rynson Lau. 2018. Modeling Fonts in Context: Font Prediction on Web Designs. *Computer Graphics Forum* 37 (10 2018), 385–395. DOI:http://dx.doi.org/10.1111/cgf.13576

[66] 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.