2

A New Tool: Research Video Martin Grödl, Moritz Resl

- 2.1 Design Decisions**
- 2.2 User Interface and Instruction Manual**
- 2.3 Pointer Feature**
- 2.4 Technical Information**

We briefly describe how the Research Video application was created, focusing in particular on assumptions, guiding principles, and design decisions. The premise was that video could be a central medium of a new form of publication, dubbed “enhanced publication.” Based on this premise, the goal was to create methods and tools for artistic and process-bound research. If possible, it would use and build upon established standards for scientific publications like the IMRaD publication structure (Intro, Methods, Results, and Discussion), references, citability, shareability, and challengeability of research results.

Since it was established that annotated video would be the main medium of the project, we conducted a cursory investigation of other major use cases of annotated video: film (subtitles, transcription), linguistics (phonetics, gestures), anthropology/ethnology (“coding”), and computer vision (automated segmentation).

Next we looked at “prior art,” that is, existing video annotation software. We tested seven applications in total, including ELAN and ANVIL, each with the same simple use case of opening a short video file and annotating a certain region. We took note of a number of important observations, which allowed us to formulate our own goals more clearly.

Guiding Observations in the Development Process

- **Technical restrictions**
Most of the tools were hard to even get running due to technical restrictions like operating system requirements or dependencies on other software or libraries.
- **Unsupported/outdated systems**
Some were based on closed or unsupported/outdated systems like Flash.
- **Lack of support**
Video format support was generally lacking.
- **Made for experts**
Most of the tools were focused on certain use cases, like transcription or linguistics, and made by experts in the respective fields for use by other experts.
- **Subpar user experience**
In general, user experience was subpar. User interfaces were laggy and unresponsive, and conventional keyboard shortcuts (for example for play/pause, copy/paste) were often unsupported.
- **Difficult to get started**
It was surprisingly difficult to get started using those programs, even with our minimal use case. In many situations it was impossible to complete the task intuitively, without studying the user manuals.

2.1 Design Decisions

Based on this research we identified a number of core concepts common to many annotation tools as well as video editing software. To facilitate communication within the project team we agreed to consistently use certain terms for these concepts. This resulted in the following nomenclature.

Terms for Core Concept

- **Timeline** (annotation board)
- **Track** (tiers, layers)
- **Annotation** (marker, interval, element, label)
- **Annotate** (coding, labeling, marking, segmentation)
- **Annotation scheme** (coding scheme, controlled vocabulary)

Taking into account the findings above, we came to the decision that a fresh take on video annotation was promising, and we formulated our goals.

Goals for the Reasearch Video Tool

- **Easy to use**
We found that most current tools are lacking in User Interface Design. Therefore we decided to make a user-centered design and usability by nonexperts our core goals.
- **Easily accessible**
As many previous tools present a significant obstacle to users by requiring specific platforms, install procedures, or additional restrictions, we aimed to remove this hurdle by building a tool on web technologies. This meant that our application should run in any current web browser, requiring no install.
- **Open**
Being part of a research project, we decided to use open technologies as much as possible as well as an open source development model. The complete source code of the application is open and can be studied, modified, and extended freely.

- **Based on best practices**

We decided to put a special emphasis on following well-established conventions regarding user interfaces and the video medium in particular.

In the following we briefly point out a few examples of guiding principles we tried to follow in the design of the Research Video application.

Examples of Guiding Principles

- **Habituation**

If something works in a certain way, it is expected to work like that everywhere and always, even across applications. When making use of this principle, the user no longer needs to think consciously about how certain things work. For example, there are commonly used hotkeys, like Spacebar for toggling between play and pause and Backspace for deleting the current selection of elements.

- **Streamlined repetition**

Refine common and repetitive tasks first, to make them as simple as possible. In the case of our application, this mainly concerns adding and editing annotations, which forms the core task of Research Video.

- **Instant gratification**

Help your users achieve a feeling of success, even if it's just small things. In our case, when somebody starts up the application for the very first time, we immediately give them a working example project to explore and play around with.

In the course of developing the application we adopted two additional principles, mainly due to limitations on development resources, but very helpful nonetheless. Already in the prototype phase of the project we decided not to include any networking component. So even though the application is accessed online, there are no servers hosting video and annotation data. The web is just a delivery mechanism for the application. All the application data is kept locally in the browser's storage and is retained between sessions. Additionally users can export their projects, producing a ZIP file that can be stored or shared with other users.

Obviously this decision saved resources for building and maintaining server components, but it proved especially practical in academic workshop settings, where many users needed to access the application over spotty Wi-Fi connections.

When developing new features in a user interface, there are usually multiple options to enable a certain user action. Often there is no clear better or preferred option; on the contrary, they often are complementary and accommodate different users. Being an experimental project with limited resources, we decided that in most cases it wasn't feasible for us to implement multiple solutions. Thus we chose to adopt as a guideline to provide one solution for one problem, sometimes called the KISS principle. This allowed us to develop features more quickly and gather user feedback before elaborating on the design.

Finally it should be noted that all mentioned design principles are not strict rules, but guidelines. Sometimes they can even be in conflict with each other, so a decision of priority has to be made. Nevertheless these guiding principles provide a valuable and consistent basis to make informed and well-balanced design decisions.

2.2 User Interface and Instruction Manual

We'll briefly describe the user interface and its functionality and features. You can watch tutorials on how to use the Research Video tool on the project website:

<https://researchvideo.zhdk.ch>

User Interface

Browser and Layout

Research Video runs in Google Chrome. The layout of Research Video is divided into four sections (A, B, C D).

A Toolbar Component

Here you can control the most important features with icons.

B Video Component

This is where the video player is located with the selected video file.

C Inspector Component

The Inspector hosts all annotations and serves as a tool for editing annotation data: Start Time, Duration, Title, and Description. A selected annotation (indicated by a frame) can be deleted with the Backspace key.

D Timeline Component

The Timeline hosts all tracks and annotations. Time is represented from left to right. Furthermore, a playhead indicates the current time position within the video file. Annotations are added here; start time and duration can be adjusted by dragging. In this component, tracks can be added, edited, and deleted.

Instruction Manual

Start Research Video

Research Video is developed for Google Chrome. Download it here:
→ <https://www.google.com/chrome>.

To start the application, visit → <https://rv.zhdk.ch>.

Load Video/ Change Video File

The first thing you likely want to do is start a project. Open the Project Overlay by selecting the respective button in the Toolbar (A) and select a video file you want to work with.

The video file should be in MP4/H.264 format.

Video Format Considerations

The best supported video format for the web browser (and therefore video) is:

- MP4 with H.264 Codec
- Recommended resolution 720p30 (1280×720 px, 30 FPS)
- Recommended target bitrate 5 Mbps
- Recommended maximum length: 60 minutes

To export this format from Adobe Premiere, use these settings:

- File → Export Media
- Format: H.264
- Preset: Mobile Device 720p HD

You can convert existing videos using the free tool Handbrake:

- Download: → <https://handbrake.fr>
- Preset: General → Fast 720p30

Set Title for Project

An arbitrary project title can be set. Saved project files are named after the title. If no title is set, it defaults to “Untitled project.” Below a cropped view of Research Video project named “Projektname1.”

Add/Edit/Delete Annotation	<p>To add an annotation, double-click within a track in the <u>Timeline</u> (D) or select the respective button in the <u>Toolbar</u> (A). A window for the annotation appears in the <u>Inspector</u> (C).</p> <p>To edit an annotation, select it either on the track or directly in the <u>Inspector</u> (C) with a mouse click. A selected annotation is highlighted with a black frame and can be edited in the <u>Inspector</u> (C).</p> <p>Selected annotations can be deleted by pressing the Backspace key or by pressing the respective button in the <u>Toolbar</u> (A).</p>
Add URL to Annotation	<p>The annotation format is minimal (text-only) but allows for linking other resources by pasting a URL into the description of an annotation. This way anything on the web can be attached freely to any annotation.</p>
Add/Edit/Delete Track	<p>Add a track by selecting the respective button in the bottom right corner of the <u>Timeline</u> (D). To delete a track, hover over the track and click the X-symbol in the top right corner. Click the title of a track to edit it.</p>
Hide Track	<p>You can hide or reveal a track by toggling the eye icon at the right side of a single track. Hiding tracks can improve the readability of the project, if you have created many tracks.</p>
Export Project	<p>Open the <u>Project Overlay</u> by selecting the respective button in the <u>Toolbar</u> (A) and select the <u>Export</u> button. Think of exporting a project as a way of saving it. You can also share the file with others. One can simply open the project in the app.</p>
Open Project	<p>Open the <u>Project Overlay</u> by selecting the respective button in the <u>Toolbar</u> (A), then select the <u>Load</u> button. You can now load previously saved *.rv files. To work on more than one project, open another window in your browser.</p>
Reset	<p>Open the <u>Project Overlay</u> by selecting the respective button in the <u>Toolbar</u> (A), then select the <u>Reset</u> button to reset the current project to its template state.</p>
Show Current Annotations Only	<p>A toggle that, when checked, hides all annotations in the Inspector that are not currently under the playhead. Works dynamically, that is, when the project is playing. →</p>

Search Function

Typing text into the search box starts a live search. The annotations that match the search are displayed in the Inspector. The chronological order of annotations in the Inspector is preserved.

There is a toggle Apply to Timeline, which, when on, also applies the search result to the Timeline. Only annotations that match the search are shown on the Timeline.

Tagging

A Tagging system helps organize, find, and view annotation content.

Within any annotation text, a tag can be added by using the hash character (#) followed by any word. Example: #experimental. This is consistent with the use of hashtags in many well-known web applications (Twitter, YouTube, Instagram, Facebook, etc.). A visual highlight (background color) is applied to tags.

After a #-character is typed, the tagging box opens, which shows a scrollable list of all used tags in this project sorted alphabetically. As you type, the list is filtered dynamically, only showing tags that begin with the typed string. When selecting Space or Enter or selecting one of the suggestions, the tag is added.

To delete a tag, hover over the tag, and an X-symbol will appear, overlaying the text after the tag. Select the X-symbol to remove the tag.

2.3 Pointer Feature

Spatial position in video can be highlighted by using a pointer. This enables the user to draw attention to specific areas of the screen, thus not only indicating when something of interest occurs but also where. This should be handy in many cases, for example, when multiple dancers, objects, props, etc. are on-screen simultaneously.

In this first iteration, we intentionally limit this feature to indicating a single point in screen space for the entire length of the annotation.

An annotation now consists of start time and duration (both timestamps), a screen location (x and y coordinates), and the annotation content (text).

The pointer extends the annotation data model by introducing an optional x/y-value pair. On an abstract level this means that in addition to referencing a specific time in the video, an annotation can now also reference space. In the context of video, space refers to the two-dimensional image plane of the frame.

Pointer Workflow

Create Pointer

Select the gray dot in the annotation (Inspector view). The dot color in Inspector changes to the respective track color. The pointer appears in the center of the video; and the playhead jumps to the beginning of the annotation.

Set X/Y Position

If the annotation is selected (indicated with a black border), the pointer can be dragged. If the pointer is selected (and possibly dragged) in the video, the annotation is also selected.

Delete/Deactivate Pointer

To delete the pointer, hover over the colored dot in the Inspector view. An X-symbol will appear. By selecting the X-symbol, the pointer disappears in all components and the dot color returns to gray.

Pointer Indication
in UI

- A

Pointer Representation in Video

Small circular shape, white stroke
If the corresponding annotation is selected: white outer glow

- B

Pointer Representation in Inspector

Colored dot: active pointer
Gray dot: no pointer

- C

Pointer Representation in Timeline

Black dot in front on text (if any)

Chronological Sorting

If there's more than one pointer at a given time, sorting should be chronological, that is, the last occurring point is drawn last, and therefore is on top.

Hotkeys	<div>Db1 Click in Timeline</div> <div>Backspace</div> <div>Spacebar</div> <div>+ or]</div> <div>Cmd + C</div> <div>Cmd + V</div> <div>Cmd + Z</div> <div>Cmd + Shift + Z</div>	<div>Add new annotation</div> <div>Delete selected annotations</div> <div>Pause/play video</div> <div>Add new track</div> <div>Copy selected annotations</div> <div>Paste annotations</div> <div>Undo</div> <div>Redo</div>
---------	---	---

2.4 Technical Information

This section provides some technical information, mainly aimed at developers who want to extend the software.

All code is free software and open source. Development takes place publicly on GitHub, where the latest version of the source code can be viewed and downloaded:

<https://github.com/StudioProcess/rvp>

The source code is distributed under the GNU General Public License, Version 3. According to this license, anyone is free to use the software for any purpose, change the software for their own needs, and share the software including any changes of their own, provided the changed software is distributed under the same terms. More information including the full text of the license is available online:

<https://www.gnu.org/licenses/gpl-3.0.en.html>

Core Technologies

The web framework used is Angular. It provides a build system (Angular CLI) as well as a modular component-based architecture for the application:

<https://angular.io>

For managing the application state we use NgRx. This technology allows us to deal with complex user interactions and their effects on the application state and enables crucial application features like Undo/Redo:

<https://ngrx.io>

Core Data Model

The core data type in the application is an annotation. It consists of an ID, so it can be uniquely referenced, a global timestamp (UTC in seconds with fractional part), a duration (seconds with fractional part), and an annotation body called “fields.” Currently only a single field called “description” is used, containing a plain text string.