Interaction Techniques for Comparing Video

Adam Baker
adam.baker@usask.ca
University of Saskatchewan
Saskatoon, SK, Canada

Carl Gutwin
carl.gutwin@usask.ca
University of Saskatchewan
Saskatoon, SK, Canada

Justin Matejka
justin.matejka@autodesk.com
Autodesk Research
Toronto, ON, Canada

Ian Stavness
ian.stavness@usask.ca
University of Saskatchewan
Saskatoon, SK, Canada

ABSTRACT

Comparison is a well-studied task in visual analytics, but there is still little support for comparison of temporal streams such as video. There are a wide range of tasks that involve video comparison, but there are very few systems or techniques to support this kind of analysis. To help address this problem, we have developed new interaction techniques that explicitly support video comparison. We provide techniques for equalizing the reference frame of videos to be compared, juxtaposition techniques for enhancing side-by-side and small-multiples comparisons, superposition techniques for comparing overlaid videos, explicit-encoding techniques that visualize differences between extracted points, and temporal-to-linear techniques that translate between a temporal sequence of frames and a 1D timeline. We built a demonstration system with five different datasets, and evaluated our interaction techniques in two ways: an analysis of steps to show their efficiency, and a preliminary user study to explore learnability, utility, and usability.

KEYWORDS

Visual comparison, video, time-lapse, interaction techniques

1 INTRODUCTION

Comparison is a ubiquitous task in visual analytics, and researchers have looked at how to support several kinds of comparisons – for example, elements in charts such as bars or lines [48], multiple instances of entire charts [22], images [8, 26], or differences between document versions [24, 45]. One type that has not received as much attention, however, is comparison of temporal streams such as video. This lack of attention is not because of a lack of use cases — people often need to compare video data. For example, a researcher who

compares the growth of two crop trials using time-lapse video; a security analyst who looks for differences in security-cam records; a coach who compares the golf swings of a novice and an expert; a basketball player who compares several videos of a particular shot to look for variance; or a baseball fan who wants to learn the differences between different kinds of pitches.

Despite the wide range of tasks that involve video comparison, there are very few systems or techniques to support this kind of analysis. Some video editors allow two videos to be played side by side, some baseball fans have made pitch overlay videos (e.g., [10]), and a few time-series datasets have been presented with a "window-blind" slider [9] – but overall, video-comparison tasks are difficult to carry out, requiring several ad-hoc and manual operations.

To help address this problem, we have developed a set of inter-action techniques that explicitly support video comparison. We designed the techniques around common video comparison tasks and organized them into five categories:

- Techniques for *equalizing the reference frame* of videos to be compared, including spatial and temporal adjustments;
- *Juxtaposition* techniques for arranging videos in side-by-side and small-multiples formats, and techniques for pointing and marking within juxtaposed videos;
- *Superposition* techniques for overlaying two or more videos, managing the stacking order, and controlling the visual dominance of the superimposed layers;
- *Explicit Encoding* techniques for visualizing the differences between key extracted points in the video;
- *Temporal-to-Linear* techniques that can translate information such as annotations between a temporal sequence of frames and a 1D timeline representation.

We built the Video Comparison Toolkit (VCT) – a demonstration system to explore and evaluate our interaction techniques with several different datasets, including time-lapse videos of crop trials and individual plant growth, videos of baseball pitches and soccer kicks, point-based traces of pollution-model output, and animations of actual and modelled sea ice extents.

We evaluated our techniques in two ways. First, we analysed the steps needed to carry out different comparison tasks with our approach versus the ad-hoc methods that would be needed with current tools. Second, we tested the techniques in a preliminary usability evaluation in which participants carried out realistic video-comparison tasks with our demonstration system. Our analytical evaluation showed that for many comparison tasks, the new tech-niques greatly reduce the number of steps required, and the user

study showed that the interaction techniques were understood and used by all participants, and that users clearly recognized the increased power and efficiency enabled by the techniques.

Our work makes three main contributions. First, we analyze fundamental tasks and operations in comparing temporal sequences such as video data. Second, we provide a suite of interaction techniques that support those tasks and operations. Third, we provide preliminary evidence that the new techniques enable improvements in efficiency for video comparison and that everyday participants can successfully understand and use the techniques. Overall, our work provides designers with new methods for enabling comparison tasks with video, which can improve the utility and usability of video and visualization tools.

2 RELATED WORK

2.1 Comparison in Information Visualization

Enabling and supporting comparison of visual information is a rich research area, although there has been substantially more work carried out for comparing static visualizations than for video or animation. In the information-visualization domain, the work of Gleicher and colleagues [18, 19, 33] is often used as an organizing framework to classify comparison techniques into three approaches: juxtaposition, superposition, and explicit encoding.

Juxtaposition arranges different visual entities in a side-by-side layout, in order to allow users to see similarities and differences across the visualizations. An important precondition for effective comparison using juxtaposition is that all visualizations use the same reference frame so that visual differences accurately reflect differences in the underlying data. Common visualization techniques that use juxtaposition are the Trellis system [5] and the small-multiples method [6]: each of the multiples has a similar layout but different data, allowing comparisons by looking across the images. Small multiples are used in many visualization scenarios such as scatterplot matrices [22], geographical propagation [42], and immersive environments [31]. Juxtaposition can also be achieved interactively: for example, the CompaRing technique brings comparison candidates close to the cursor on selection [51].

Superposition involves putting two visualizations in the same spatial location so that differences are visible in the same reference frame. This allows similarities and differences to be seen more clearly, but also introduces the problem of clutter – some representations such as space-filling methods or dense scatterplots do not work well as overlays, and this can be a problem for overlaying video (although filtering techniques and transparency can be used to reduce occlusion [32, 34], and techniques such as "shine-through" representations can highlight differences [32, 52]).

Explicit encoding creates and visualizes a secondary dataset that specifies differences between two primary datasets. Many types of explicit encoding are possible: for example, showing the existence of differences, the magnitude of differences, or the type of differences (limited only by the ways in which two datasets can be compared) [41]. Researchers have demonstrated several explicit-encoding methods in visualization research, including "diffing" systems that show changes between versions of a document [7, 24, 40, 45, 49], "diff matrices" that show a matrix of line pairs [47], or differences between tables at different times [41].

Researchers have also extended Gleicher's three basic categories to include other representations. Different visualizations can be presented sequentially in the same location, either using the idea of Rapid Serial Visual Presentation (RSVP) [4], or using animation to smoothly morph from one dataset to another [15] (a combination of temporal juxtaposition and superposition). Other techniques provide interactive access to the bottom image in a superimposed pair – e.g., by allowing the user to 'peel back' the top image [52], or by providing a 'window-blind' slider that controls visibility [9]. Other researchers have extended the idea of juxtaposition through nesting [28], and have added some elements of superposition to juxtaposed visualizations by overloading one representation with details from another, such as showing elements that are present in one visualization but not in another [27].

2.2 Video Exploration and Manipulation

Techniques for working with video and other linear temporal data have also been well studied in HCI [46]. Researchers have looked at many topics including video annotation [11], methods for exploring large collections of videos [37], exploration and visualization of surveillance video [23, 38], navigation through direct video manipulation and interactive video summarization and retrieval [14], techniques for working with video summaries of meetings and events [29], methods for managing live video streams [50], and ways of improving interactive performance for network-bound online video (e.g., [21, 25, 35, 36]). In addition, researchers have also looked at interaction techniques for other types of temporal data. For example, Grossman and colleagues developed techniques for working with event streams that can be played like video, such as the history of a user's interaction with a document [20].

However, there is less research investigating comparison of video. A few projects have created prototypes that display many videos at once – for example, Murphy's "A History of the Sky" [39] which showed 365 videos of the sky, each taken on a different day; or Vuillemot's "Watching +100 The Simpsons Episodes" [56] which showed differences in the intro sequences for the Simpsons cartoon. In addition, Chen and colleagues developed methods for comparing individual video frames to find differences [8].

One type of linear-media comparison that has been examined in HCI, however, is the overlay technique. Several HCI researchers have investigated the idea of overlaying two streams of visual information, often with the goal of helping a user understand or improve movement. Bau and Mackay's Octopocus system explored overlay-based "feedforward" for training users with a gesture vocabulary [3], and other systems such as Arpège [17], ShadowGuides [16], G3 [13], and Gestu-Wan [44] follow similar feedforward principles to help users learn input actions. This idea has also been expanded to full-body movement, with augmented reality systems that provide feedback on movement training. A reference (e.g., dots, a stick figure, or video of an expert) is overlaid on video of a learner's actions to help them improve [2, 12, 53].

In addition, Cherukuru and colleagues extended the "window-blind" interaction method (often used to compare before-and-after images) to the comparison of animations – their system allows the user to compare animations of modelled and actual sea ice in the Arctic [9]. Finally, a recent system overlays multiple videos

of baseball pitches to compare movement of the ball [10], a task that is relevant to what we investigate here. However, the system does not look at techniques for interacting with the overlays, but rather focuses on how to automate the process of creating a pitch-comparison video, and is primarily concerned with image-processing methods to accurately track the ball in the video.

3 FRAMEWORK: VIDEO COMPARISON TASKS

There are many ways in which users may want to compare video data. Based on previous work on low-level comparison tasks in information visualization (e.g., [1, 43]) and our experiences working with domain experts in plant science who need to compare time-lapse sequences of crop trials, we have identified a set of core user questions and comparison tasks, and use those tasks as a guide for designing our techniques.

3.1 User Questions

Given a set of videos, there are several questions that the user might ask as the starting point for a comparison task:

- *Do things appear in one but not the other?* This question considers differences in the existence of objects, movements, or changes.
- *Are things the same or different?* For videos that contain comparable objects or actions, this question may focus on spatial variables (i.e., movement) or visual variables (e.g., colour, size, shape) over time.
- *How large is the difference?* If motions or visual variables differ across two videos, how large are the differences?
- *Where and when are the differences?* Differences in existence, movement, or visual variables can occur at different spatial or temporal locations within the reference frame.
- *In which video does X happen first (and how much earlier)?* This question involves the relative occurrence of domain-specific events such as specific motion patterns, changes to a visual variable, or appearance or disappearance of an entity.

3.2 Comparison Tasks

From the user questions above, as well as task and interaction typologies developed for information visualization (e.g., [1, 43]), we identified three comparison tasks with specific use cases.

- **U1**. *Identifying relative temporal occurrences*: e.g., a plant scientist assesses the relative flowering times of time-lapse video of two test crops.
- **U2**. *Comparing two kinds of motion to understand the difference*: e.g., a baseball fan watches videos of a curveball and a slider to try and understand the differences.
- **U3**. *Finding differences in visual variables such as size, colour, pattern, location, or shape*: e.g., a climate scientist might compare animations of yearly sea ice to determine which year has a larger ice area.

4 TECHNIQUES FOR VIDEO COMPARISON

We organize our video-comparison techniques into four categories: basic techniques for standard manipulations; techniques for equalizing the reference frame; techniques for the three categories proposed by Gleicher (juxtaposition, superposition, and explicit encoding); and a final category with techniques that translate between temporal and linear representations.

4.1 Demo System and Basic Techniques

All of the techniques are implemented in a web-based demonstration application (Figure 1). In this system, each video loaded into the system is shown as an object in a visual workspace – the video is placed in a frame that allows selection, dragging, and resizing, and the frame also contains a local control bar that includes a local timeline controller and endpoint markers for the video.

Video playback is controlled either through the individual timeline controllers or through a global scrubber widget that is available at the bottom of the workspace (allowing simultaneous scrubbing of all videos). Videos can be scaled, zoomed, and panned to allow different videos to be equalized for comparison; for example, Figure 2 shows a scaling-and-panning manipulation in which the user selects two equivalent points in two videos, and the system determines a transformation matrix that will equalize the zoom and pan of the two videos. When a dataset includes timing information, users can toggle the timestamps on and off to provide further contextual information. Users can also save a snapshot of the configuration parameters (e.g., location, size, zoom, play location, and overlay state of a collection of videos); these snapshots can be reviewed and revisited using a menu in the system's UI.

4.2 Techniques for Equalizing Reference Frames

The first task in comparing two videos is ensuring that the reference frames are equivalent. In addition to the basic techniques described above for resizing, zooming, and panning individual videos, we developed three augmented techniques that enable new interactions.

4.2.1 Global Transforms on Multiple Videos. Comparisons involving multiple videos that require repetitive and uniform reference-frame adjustments (U2) can become tedious. To simplify this task, we extended our basic techniques (zoom, pan, and coincident-points registration) to work with multiple videos at once. When multiple videos are selected, the user's manipulation on any video will be applied to all selected videos when the Control key is pressed.

4.2.2 Overlay Feedback for Zoom and Pan. Equalizing the reference frame can be easier when two videos are transparently superimposed. This allows invariant elements of the two videos to be seen in the same spatial context (U2). Once the registration process is complete, the video can be left in a superimposed arrangement, or dragged back to the workspace for use in a juxtaposed comparison.

4.2.3 Endpoint Adjustment and Normalization. Unlike static visualizations such as charts and images, videos contain temporal attributes such as extent and frame rate that can cause desynchronization between multiple videos. Therefore, in addition to equalizing the spatial reference frame, techniques are also needed to equalize the temporal reference frame of the videos.
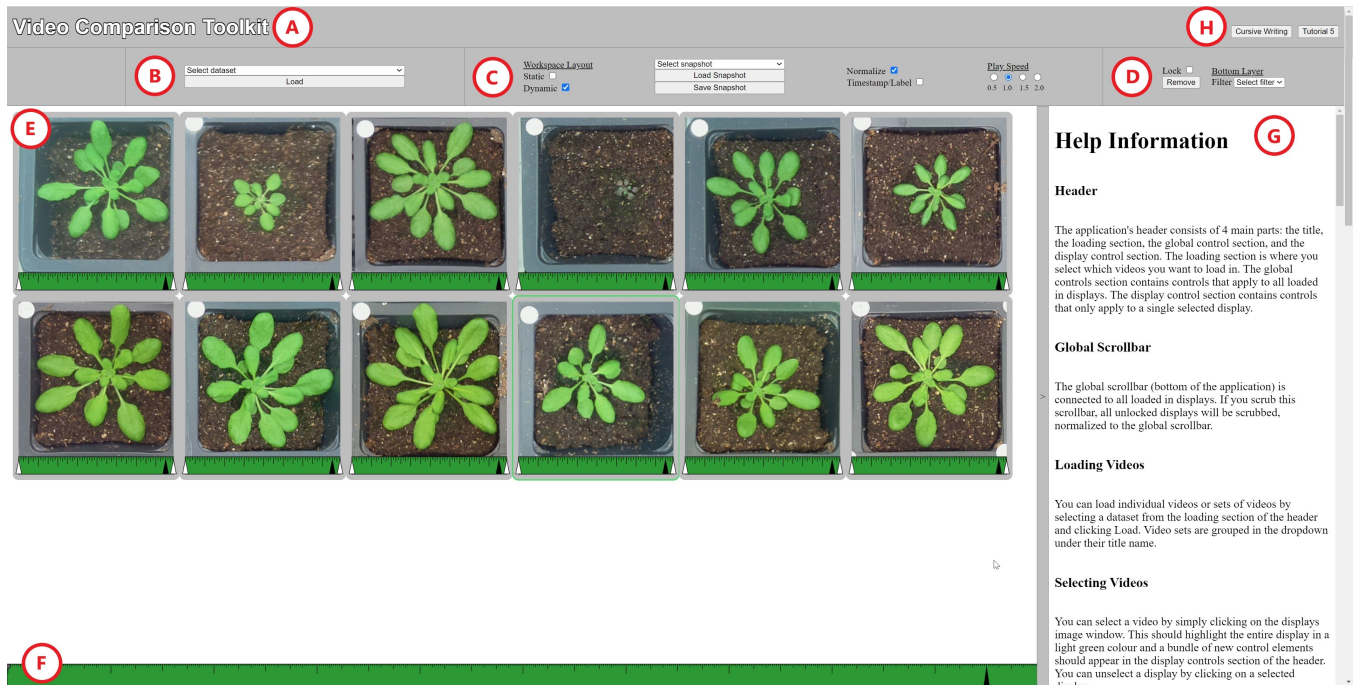
Figure 1: Web-based demonstration application. (A) Title (B) Video selection dropdown (C) Global system options (D) Individual video options (E) Workspace with 12 plant videos (F) Global timeline widget (G) Help tab (H) Links to examples.
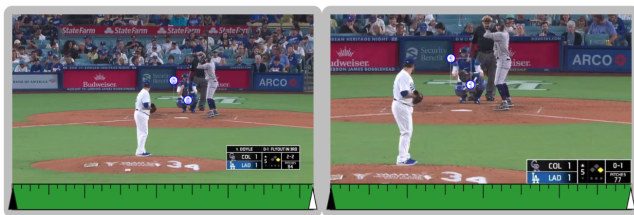
(a) Before transformation (corners of the strike zone have been selected in both videos).
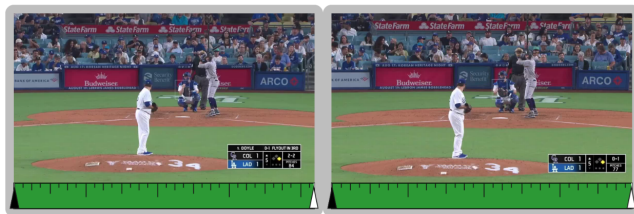
(b) Equalized videos after transformation.

Figure 2: Coincident-points technique. Videos are misaligned (a), so points are placed to create an alignment transformation and equalize the videos (b).

The first technique involves endpoint adjustment. Videos can differ in length, and may only contain a portion of the events in another video, but comparing video requires that all videos are showing the same thing at the same time. Our technique adds two endpoint markers to the local control bar of each video, and these can be dragged to adjust the start and end frames of the video. For user task U1, the flowering cycles of different crop trials may start and end at different times, which makes comparison of flowering time difficult for a crop breeder. The user can adjust the endpoints of the video by moving the markers in the control bar to start each video at the beginning of the flowering cycle.

Figure 3: Endpoint adjustment: time-lapse videos of two Canola varieties, with endpoints adjusted to show flowering.

The second technique allows the user to normalize the playback rate of several videos – needed when videos have different frame rates, or when events to be compared have different lengths. For example, comparing the relative position of two pitches that have different speeds is difficult because the ball position differs in each video. Users can normalize the rates of compared videos by toggling a mode in the UI (Figure 1c); when videos are played together, their playback rates are normalized using a ratio equation, ensuring that they have consistent start, middle, and end. We note that playback

should not always be normalized – absolute playback is important when phenomena take different amounts of time.

4.3 Techniques for Juxtaposed Comparison

In addition to the basic visual workspace that allows users to arrange videos as needed, we have designed two specific techniques that further support juxtaposed comparisons.

4.3.1 Small-Multiples Grid Layout. Similar to the technique frequently used in visualization systems, our small-multiples technique places a set of videos in a grid layout to enable side-by-side comparisons of multiple datasets at once. The global scrubber widget allows the user to move through all of the videos in the grid at once (local controls work as well). We have extended the basic small-multiples idea in four ways:

- *Resizable and reconfigurable grid.* When loaded, videos are sized to fit the grid, but the grid can be resized by dragging a handle on the grid lines, which also resizes all videos. The grid is initially configured to fit the size of the workspace and number of videos (Figure 4); the dimensions of the grid can also be changed using a control panel.
- *Hiding video borders.* To reduce visual clutter for making juxtaposed comparisons, the video borders and local controls can be toggled on and off, as seen in Figure 5.
- *Position rearrangement.* Videos can be dragged around in the grid; dropping a video in a new location swaps the dropped video with the existing one.
- *Group loading* Although videos can be loaded individually into the grid, we also provide group loading that will import all videos contained in a specified folder.

The small-multiples technique works well with either a small or large number of comparators. Although detail is reduced with larger sets of videos due to the reduced size of each item, these configurations allow the observation of differences in visual variables (U3), as shown in the small-multiple arrangement of Figure 4. With the video borders toggled off, this technique also allows for the recreation of previous demonstrations such as Murphy's "A History of the Sky" [39] or Vuillemot's composite Simpsons' episode [56].
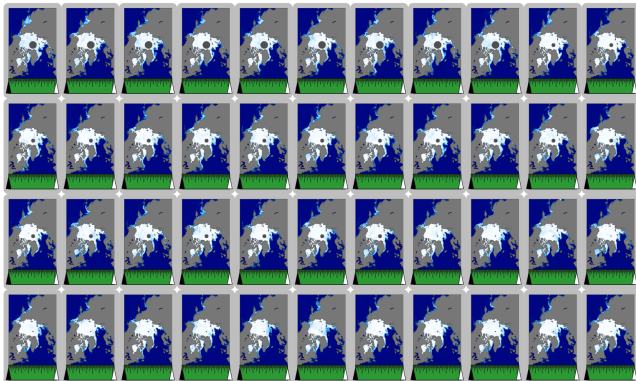
Figure 4: Small-multiples technique showing animations of sea ice changes, 1979-2022; each video is made up of 365 satellite images.
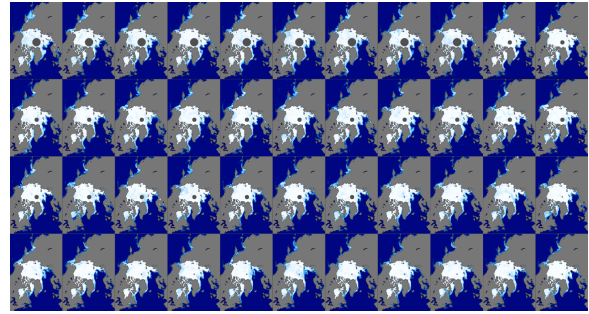
Figure 5: Small-multiples technique with hidden borders.
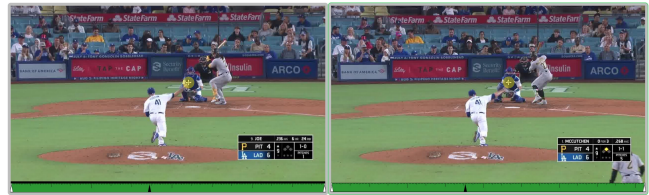
Figure 6: Shadow markers to compare different pitches from the same pitcher. A marker (yellow circle) placed in the left video is replicated in the right video.

4.3.2 Shadow Markers. A limitation of juxtaposition is that it can be difficult to visually track equivalent spatial locations (U3) across different videos, particularly if there are few consistent landmarks in the background. This problem can often be solved by using superposition, but we have also designed a mechanism for marking a spatial location across all videos. The shadow markers technique adds a visual mark at the user's click location, and the mark is duplicated on all videos in the set (e.g., across a small-multiples display). The marker becomes an artificial landmark [55] that can be used as a reference point across different videos, assisting the user in evaluating object positions different views.

Shadow markers are also useful for comparing object locations at significant events (U1) or the motion of an object over a period of time (U2). These comparisons are difficult in a standard juxtaposed layout because it is hard to precisely compare specific spatial locations in two videos, even if they have a common reference frame. By placing shadow markers at specific locations, users can accurately judge differences in position or motion (Figure 6).

4.4 Techniques for Superposition Comparison

Superposition techniques are initiated in our system by dragging one video on top of another (which creates a copy of the video to ensure that the original is unaffected by any changes). We provide two basic manipulation techniques for overlaid videos. First, transparency can be either controlled manually for each video, or can be set automatically using a harmonic series (i.e., the bottom video is fully opaque, and the opacities of successive layers are 1/2, 1/3, 1/4, etc.) – this ensures that all layers have equal visibility [34]. Second, users can arrange the stacking order of the videos using a simple cycling mechanism attached to a hotkey that moves the current top

video up or down in the stack. In addition to these basic capabilities, we developed the three extended techniques described below.
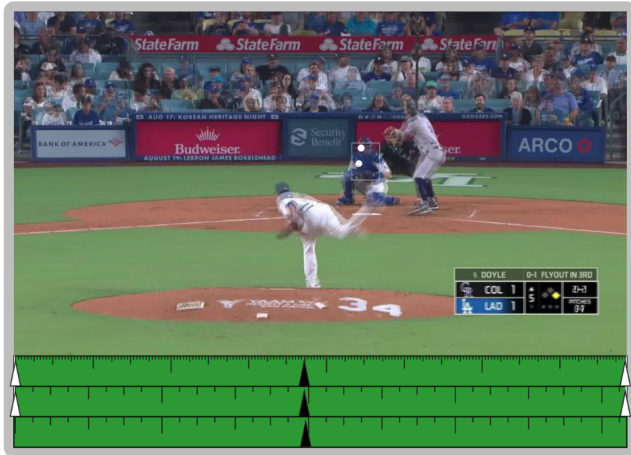

Figure 7: Superposition video of two pitches from the same pitcher. (Ball brightness has been increased for visibility).

4.4.1 Window-Blind Sliders. There are some situations in which transparency is not suitable for comparison because details in each video needs to be seen clearly (U3). In these cases, other mechanisms can be used to manipulate which video is visible. Previous work has demonstrated a slider-based technique in which the spatial frame is divided between two images, with the slider controlling the division line (e.g., [9]). We have replicated and extended this technique, and combined it with layer cycling described above.

When the window-blind mode is activated, the overlay pane is split into two parts with different videos (at 100% opacity) on each side. A slider is added to control the relative size of the two parts, and the user drags the slider back and forth to compare elements of the videos that are in the same position. The slider can be horizontal or vertical to allow comparison of different parts of the overlay.

In Figure 8, the technique is used to compare traffic levels during summer and winter. The technique can be combined with adjustable opacity levels and layer cycling – for example, pairs of transparency-stacked videos can be compared using the window blind.

4.4.2 See-Through Lens. We extended the idea of the window-blind slider to allow the user to see through the top video in a particular region. When invoked with a hotkey, a see-through lens appears in the middle of the superposition video, and shows the underlying video (both videos are 100% opaque). The user can adjust the position of the lens by dragging. The see-through lens is useful in situations where video images are similar and where transparency would lead to unwanted visual clutter – for example, in U1, comparing the time at which two crop varieties begin flowering is difficult when using transparency, because the details in each layer blend together. Using the see-through lens, the user is able to clearly view the lower layer, and more accurately see which flowering cycle begins first (Figure 9).
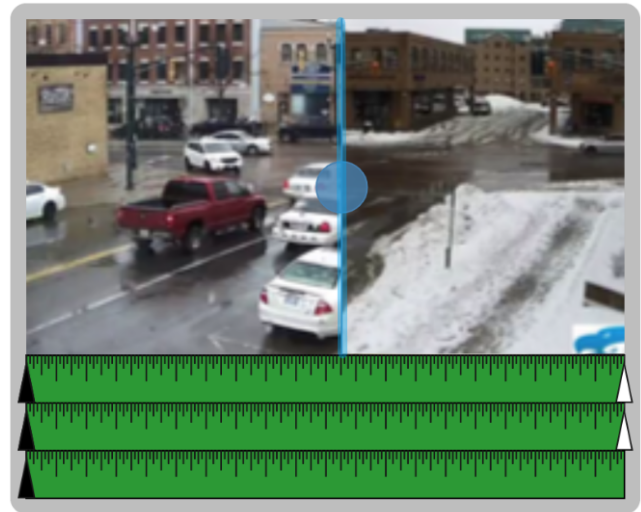
Figure 8: Window-blind overlay showing a horizontal slider, used to compare summer/winter traffic levels.
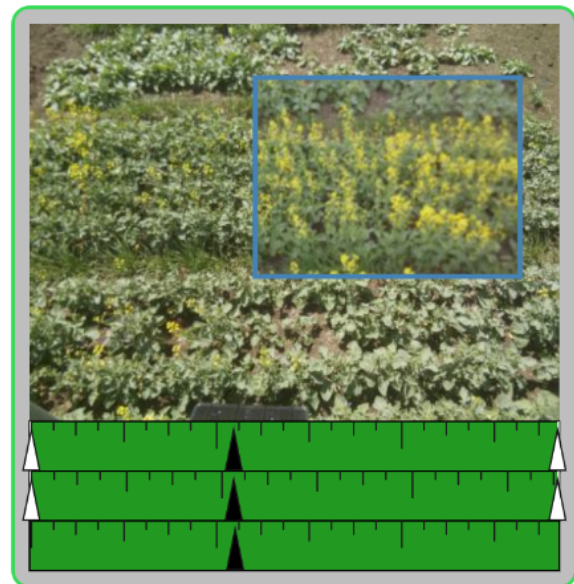
Figure 9: See-through lens showing a specific region in the underlying video at full opacity.

4.4.3 Image Filters. Visual filters can assist overlay comparisons by increasing contrast between two videos or highlighting important elements; however, real-time filtering creates a substantial delay in the comparison task. Therefore, when a new video is added to the demonstration system, we run a set of offline preprocessing steps to create several alternate versions of the video (including grayscale conversion, background subtraction, and image segmentation). When the video is loaded into the system, these alternate versions are available through a menu. For example, Figure 10 shows

time-lapse videos of Komatsuna plants that include a segmented alternate [54].
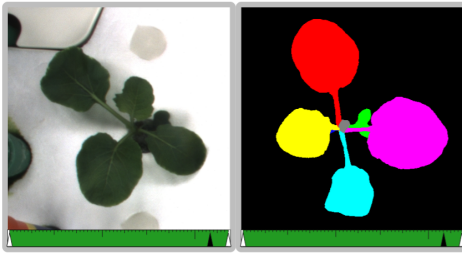
(a) Komatsuna plant(left) and segmented plant (right).
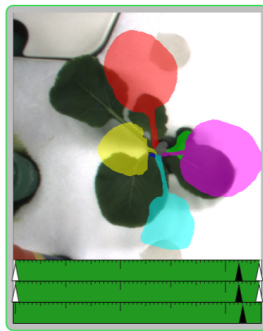
(b) Combined images in an overlay.

Figure 10: Overlay with image filters, showing one RGB plant image and one segmented image. Image source.

4.5 Techniques for Explicit-Encoding Comparison

We have developed a technique to enable explicit-encoding overlay comparisons. Explicit encodings require that position data has been generated by an external program, or extracted from the source video (either manually or using a segmentation algorithm such as Segment Anything [30]). Our technique operates downstream of this process, and assumes that two input files are available that consist of X,Y locations and timestamps. Our technique creates a video object for this pair of input files, and can then visualize the points in several ways: simply drawing the points over time, connecting the points, or drawing lines between equivalent points to indicate the distance between them. Each file's points are drawn in a different colour. Figure 11 shows an example from an environmental-cleanup scenario. The visualization shows two outlines – one indicating the predicted underground extent of a pollutant that is being reduced over time by a bacterial attenuator, and one showing the actual extents based on sensor readings. The two outlines are compared using two different visualization methods. User task U2 represents another common use case, in which the X,Y locations along a baseball's path can be collected and compared.
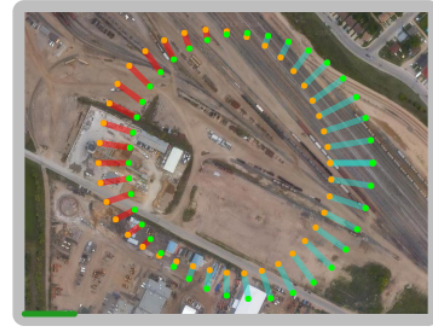
(a) Difference lines visualization.
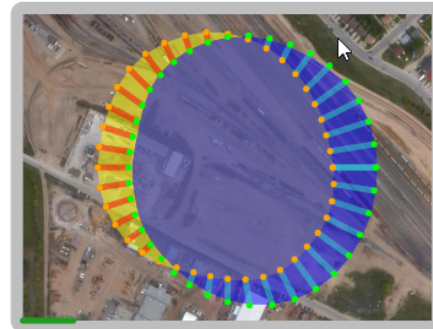
(b) Difference lines plus area visualization.

Figure 11: Explicit encodings displaying predicted changes to a pollution spill. Difference lines and areas represent the relationship between predicted and observed point data.

4.6 Techniques for Translating Between Temporal and Linear Representations

We developed two techniques that translate between a streamed temporal video and a static linear representation of the video's time span. We overload the video timeline control (which naturally represents the full time span of each video) with additional visual elements, allowing users to make comparisons in the linear space without moving through the frames of the videos.

4.6.1 Timeline Annotations. When the user observes events of interest within a video (U1), they can mark that position in a specific colour by pressing the 1-8 number keys. The annotations are saved and displayed within each timeline as coloured lines; the annotations can be used as bookmarks for quick navigation (clicking on an annotation mark snaps the video to that position), but can also be used for comparison tasks. When two videos are overlaid, both of their annotation tracks are shown in the composite timeline – if the user has marked events that are common to both videos, they can quickly see the relative temporal difference between these events.

For example, markers for events in the wind-up and pitch of four videos have been marked in Figure 12, and the difference between these events can be clearly seen in the combined timeline.

4.6.2 Annotation-based Synchronization. The timeline view can also be used to synchronize videos for playback from a particular
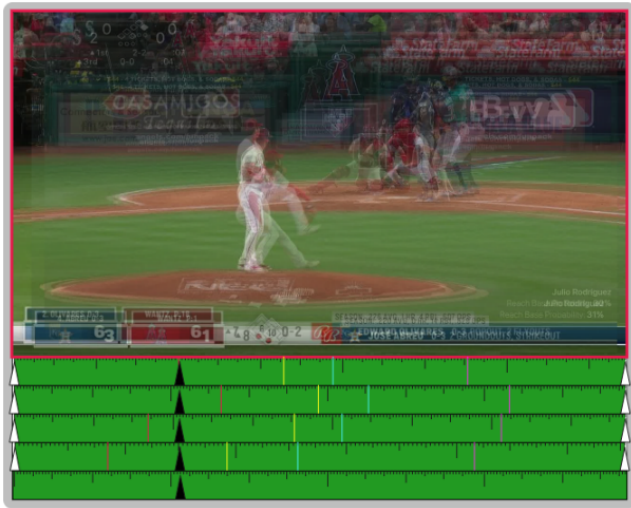
Figure 12: Annotations. Four overlaid pitch videos, with timeline annotations for beginning of the wind-up (red), release of the ball (yellow), catch by catcher (cyan), and call from the umpire (magenta). The red border indicates that one of the videos has reached the corresponding marker.

event that has been marked with annotations in different videos. Clicking an event marker in the timeline snaps the video to that location, and doing this in multiple videos brings them all to that event. Videos can then be played forward from the point of the event – for example, to see what happened to two Canola varieties after the application of a particular fertilizer, or to see their response to a heavy rainfall (which could happen at different times if the plots are in different geographical locations). Annotation-based synchronization is a more flexible version of the endpoint manipulation described above, because it still allows viewing of the entire video, and allows the user to quickly synchronize to different points of interest, as long as these have been marked with annotations.

4.6.3 Timeline Measurement Tool. To help users compare time spans for two videos (U1), we added a measurement tool to the timeline widget. When the user presses Control and then drags in a video's timeline, the system displays the total time (in the scale of the timeline) from the start of the drag action to the current drag position. This allows users to measure the elapsed time between annotations in the timeline. An example of this can be seen in Figure 13, where the total flowering time of a Canola video can be compared with other videos.

5 ANALYTICAL EVALUATION

We carried out an analytical evaluation comparing our techniques to DaVinci Resolve, a popular video editor. We examined four basic comparison tasks (based on the tasks discussed in Section 3) in both DaVinci and our system, and recorded the number of steps required to complete each task.

- *Juxtaposition comparison*: place two videos of growing Canola crops side-by-side and compare the full extent of each video, noting similar and unique temporal events (U1).

Figure 13: Timeline measurement tool, measuring from the start to end of the flowering cycle (marked with annotations) in Canola.

- *Create a small-multiples grid*: form a 3x3 small-multiples grid of ice-extent videos that can be played simultaneously to enable comparisons of visual variables in all videos (U3).
- *Superposition comparison*: overlay two videos of baseball pitches and compare the motion of the ball (U2).
- *Equalize reference frames*: given two videos that are misaligned and scaled differently, perform transformations to equalize the two frames of reference.

The number of steps required to complete each task is shown in Table 1. Results indicate that in each task, our techniques substantially reduced the number of steps required. This is not a surprising result, given that we designed the techniques specifically to carry out these tasks – but it does indicate the degree of manual operation that is required using current tools. Positioning and equalization of videos to facilitate comparison within DaVinci and other video editors requires several setup transformations that our system performs automatically (such as juxtaposed grid positioning, drag-and-drop superposition combinations, and coincident point transformations to equalize video reference frames).

Task	DaVinci Resolve	VCT
Juxtaposition comparison	9	2
Create small-multiples grid	32	4
Superposition comparison	7	2
Equalize reference frames	16	8

Table 1: Number of steps required for four tasks in DaVinci Resolve and our Video Comparison Toolkit (VCT).

Solutions to each task were found by exploring the DaVinci documentation. We simulated a typical novice user attempting each comparison task within the video editor – although it is possible that the number of steps in our analysis is not the optimal number, it does represent what would be done by a non-specialist user. Future

studies will also consider optimal walkthroughs in DaVinci and other video editing and comparison tools at the expert level.

6 PRELIMINARY USABILITY EVALUATION

The novelty of the techniques and lack of software support for video stream comparison made it difficult to run a full comparative evaluation. As a preliminary step, we opted for a task-based evaluation to explore the learnability, utility, and usability of the techniques. Future work may provide opportunities to compare the techniques against state-of-the-art tools in a more rigorous manner. We had both summative and formative goals for the evaluation: for the summative assessment, we wanted to determine if users could understand and use the interaction techniques to complete comparison tasks, and whether they saw value in them; for the formative analysis, we wanted to know where the techniques worked well and where they failed, and what features could be added or changed to provide better support.

6.1 Video Domains and Tasks

Study tasks focused on comparing sets of short videos pre-loaded into the system. We used videos from five separate domains in the evaluation: top-down plant growth (e.g., Figure 10), crop flowering cycles (e.g., Figure 3), baseball pitching (e.g., Figure 7), arctic ice formation (e.g., Figure 4), and modelled/observed effect of hydrocarbon-eating bacteria on an oil spill (e.g., Figure 11). For each task, participants followed a general outline of what actions to carry out, but they were also allowed to use other techniques.

6.1.1 Comparing top-down Arabidopsis plant growth. The participants' first task was to observe the plant growth cycle of six Arabidopsis plants for differences in phenotypic traits such as size and shape (similar to U3). The videos were scaled to take up more than the user's screen, which meant that they had to scroll to see all of the videos. Participants performed the following steps:

(1) *Reconfigure layout*: scale video displays to all fit within the view of the screen.
(2) *Watch videos*: scrub/play through the six plant videos individually and globally.
(3) *Find details*: identify visible similarities and differences among the set.
(4) *Rank plant growth*: reposition the videos in a ranking based on a size criterion (i.e., smallest to largest).

6.1.2 Comparing Canola flowering stages. The second task involved locating and observing significant events within videos (U1). In this case, participants focused on the flowering cycles of two Canola crops. Participants performed the following steps:

(1) *Locate events*: scrub through each video (locally or globally) to find the start and end of each flowering cycle.
(2) *Trim videos*: set endpoints at the beginning and end of each crop's flowering stage.
(3) *Find details*: identify which crop flowered first and which crop finished flowering first.

6.1.3 Comparing baseball pitch path. The third task asked participants to identify differences in the pitches of two baseball players

(U2). Pitchers add spin to the ball to provide movement in the endpoint of the ball's path; the horizontal and vertical shift in the ball's flight path due to this spin is referred to as *break*. Pitchers with large break differentials between pitch types are highly sought-after for their ability to make the batter miss. Video comparison is a commonly used tool to identify pitchers with outstanding breaks.

Four videos were loaded into the system for a similar comparison (two videos for each pitcher), and misaligned by default. Participants performed the following steps:

(1) *Align videos*: equalize the reference frames of each pair of baseball videos. This could be accomplished using transformations such as zoom, pan, or coincident points.
(2) *Create overlays*: drag and drop video pairings on top of one another to form overlaid videos.
(3) *Analyze & compare pitchers*: scrub through the overlaid videos and determine which of the two pitchers had a more consistent pitch and who had a larger break in the ball.

6.1.4 Comparing arctic ice formations. The fourth task provided an opportunity for participants to work with a much larger dataset. 40 videos of daily arctic satellite imagery were loaded into the application for participants to observe trends and differences in the data (e.g., variation in ice density and extent) (U3). Increasing ice density was represented by a blue-to-white colour scale. Participants performed the following steps:

(1) *Search for trends*: scrub through all 40 videos simultaneously and look for consistent sea ice trends that occur year-to-year.
(2) *Identify significant areas*: search for particular areas on the map where change occurs. This could be accomplished using a standard visual search, or by using shadow markers.

6.1.5 Visualizing hydrocarbon-eating bacteria simulations. The purpose of the final task was to compare the utility of several of the techniques discussed above (juxtaposition, superposition, and explicit encoding). A tutorial showcased simulations of hydrocarbon-eating bacteria, with data presented as points and areas overlaid on a map of an oil spill location. Participants were asked to compare the effectiveness of the bacteria in the measured versus the simulation data. The techniques were presented as follows:

(1) *Juxtaposition*: areas and points for the measured and simulated data were visualized in two frames placed side-by-side for comparison.
(2) *Superposition*: the data was placed on top of one another for comparison.
(3) *Limited superposition*: only the points from both datasets were shown as an overlay.
(4) *Explicit Encoding*: the calculated difference between the measured and simulated data was shown using the difference lines technique (Figure 11).

6.2 Participants and Procedure

Eleven participants were recruited from a local university (mean age = 24, SD = 4.17). Participants were all experienced with desktop computer software (mean 41.9 hours/week on PCs and mobile devices), but had limited experience editing, comparing, and manipulating videos (mean 1.4 hours/week).

Participants were given the option to have their session in-person (n=3) or remotely (n=8). For in-person sessions, participants were provided a laptop that recorded the session. The experimenter remained in the room throughout the experiment to answer any questions participants had about the task or the system. Remote sessions were held on Zoom and recorded using its built-in recording feature. The experimenter remained in the conference throughout the session to answer questions.

In each 45-minute session, participants were given an introduction to the study and completed informed consent and demographics forms. They were given an overview of the toolkit and a demonstration of the techniques they would be using before beginning the tutorials. They then carried out the tasks described above. Participants were free to complete each task using any methods they wished, but steps were provided to guidance. Participants were encouraged to think aloud as they completed each task, and questions were embedded into each tutorial for the participants to answer. At the end of the session, participants completed a final questionnaire to gauge the difficulty of accomplishing each task and the usability of each interaction technique. After each session, recordings were reviewed and transcribed by the experimenter.

6.3   Results

The goals of the evaluation were to determine if non-specialist users could understand and utilize the techniques to complete tasks in real-world scenarios, and to gather information on how to improve on the system (such as usability problems, features that could be improved, and new features that would help the participants to accomplish the task). Overall, all tasks were completed by all participants, and only a few minor problems were reported. First, the demonstration videos took longer to load than anticipated, hindering the participants' ability to begin the tasks. This problem was rectified in later sessions by having participants work through each task while loading in the subsequent videos. Second, lighting in the Canola crop videos made it difficult to pick out flowers from sunlight reflecting off the leaves. Third, for the first participant, techniques that made use of the middle mouse button (such as zoom and pan) did not work correctly when used with a touchpad. We updated the system to use a hotkey for later participants.

Participants were asked about their experiences with the interaction techniques during the sessions. User feedback resulted in clear trends and provided helpful suggestions:

- Global playback was more useful for comparing multiple videos, while individual playback was preferred for finding significant details in certain sequences.
- Endpoint adjustment was deemed as not a "make or break" technique, but the majority of participants agreed that in many situations, endpoints were important. One participant gave this example: "In lots of videos there are a lot of parts that are not really important, so with a subset version of it you can go straight to the important things".
- While the coincident points technique was intended to support juxtaposition, users found that it worked well for superposition by aligning the reference frames before videos were overlaid (e.g., one participant said that "the point alignment compliments the overlay").

- Shadow markers helped participants focus on areas of interest. One participant noted that "more than anything it's helpful for a gauge of the best size... for example, the first and the last [video] looked pretty similar when you're just looking with your eyes. But once you've got the marker on you can see how big of a difference it actually made."

After the session, participants were asked to rate how difficult it was to perform tasks within each tutorial using 7-point semantic-anchor questions (1 = "Very Difficult," 7 = "Very Easy"). Overall, participants found the majority of the tasks to be straightforward, as shown in Table 2. Participants were also asked to rate the usefulness of each technique using 5-point semantic-anchor questions (from "Not Useful" to "Very Useful"). Results shown in Table 3 indicate that participants found the techniques to be useful in completing the tasks.

Task	Median	Mean
Compare juxtaposed videos	6	5.18
Play through individual videos	6	6.27
Play through all videos	6	6.27
Scale videos	6	5.91
Reposition videos	6	6.00
Trim videos	6	6.00
Video alignment	5	5.00
Overlaying videos	6	5.91
Compare overlaid video	5	5.18
Notice details across videos	5	4.55

Table 2: Post-session questionnaire responses - task-based technique usability (1-7 scale, Very Difficult to Very Easy).

Interaction Technique	Median	Mean
Small-multiple grid layout	4	4.18
Individual scrubbing	4	4.36
Individual playback	5	4.45
Global scrubbing	5	4.64
Global playback	4	4.91
Repositioning videos	4	4.00
Zooming	4	3.82
Panning	4	3.45
Timestamps	4	4.00
Endpoint adjustment	4	4.00
Normalization	4	3.82
Coincident points	5	4.55
Deleting videos	4	3.45
Overlays	4	4.18
Shadow markers	4	3.82
Difference lines	4	3.73

Table 3: Post-session questionnaire responses - technique utility (1-5 scale, Not Useful to Very Useful).

Although our evaluation only involved some of the interaction techniques described above, it does indicate that users can perform basic comparison tasks using the techniques, even with minimal video-analysis experience.

7 DISCUSSION

In the following paragraphs we consider the overall success of our techniques, review the novel aspects that can be generalized to other comparison systems, and discuss limitations of the work and opportunities for further research.

The problem that motivated this research was that comparing video currently requires the user to undertake a series of ad hoc and often laborious actions to set up the videos and compare them. Many of the interaction techniques we developed provide capabilities and functionality that do not exist in other tools or other research systems (e.g., simple creation of video small multiples, drag-and-drop overlay creation, shadow markers, see-through lenses, timeline measurement, and configuration snapshots), and others contain improvements or adaptations to methods that currently exist (e.g., lightweight endpoint adjustment, custom annotations, or multi-video zoom and pan). Overall, the techniques presented here provide a suite of capabilities that simplify existing tasks and enable new kinds of comparison. Our preliminary evaluation provides added evidence that the techniques aid non-specialist users in carrying out comparison tasks that would be difficult using standard tools.

In addition, some of our techniques involve novel ideas that could be translated back to comparison tasks in other visualization scenarios. For example, the shadow markers technique could enable new analyses in static small-multiples visualizations, in which the spatial separation of the different images can make comparisons difficult, even when the reference frame is equalized. Similarly, a see-through lens could be used in side-by-side comparisons to provide some of the registration benefits of an overlay for specific tasks, but without cluttering the display with all of the details of the other visualization.

There are still several limitations in our techniques and in the demonstration system. The techniques were developed as individual methods, and lack a consistent approach that would allow multiple techniques to be used together in a seamless fashion. The techniques are currently heavily reliant on keyboard commands and modes, and in future work we will investigate additional ways of invoking and transitioning between techniques so that they work well with each other. In addition, the web-based demonstration application lacks polish, and needs to be improved in terms of loading time, feedback, and overall UI design. Although it was usable for the eleven people who participated in our evaluation, we plan to improve it to the point where it can serve as a robust sandbox for testing our current techniques as well as new ideas. A main reason to have a usable demonstration system is that we can then carry out more detailed evaluations with domain experts who need to carry out video-comparison tasks as part of their everyday work. For example, we are working with plant breeders and agronomists who need to compare test plots of different crop varieties (e.g., the Canola examples shown in Figures 3 and 9). These scientists use video to make decisions about the success of particular treatments or varieties, and future studies will test whether the interaction techniques can support detailed real-world comparison tasks.

Related limitations in this work stem from the lack of a rigorous evaluation methodology. Our current evaluation only established that the techniques were learnable and usable, but did not investigate expert usage scenarios. Although the initial focus was on establishing video comparison techniques that were simple and intuitive, the value of these techniques has yet to be properly evaluated. Future studies will focus on robust investigations into each technique and how they perform compared to alternative video editing and comparison tools such as AVISynth and VideoCompare for insight into the utility of the toolkit at an expert level.

There are also several other avenues for future research in this area. First, we plan to integrate techniques from systems that allow exploration of larger collections of videos (e.g., [37]), to investigate methods for combining overviews with comparison capabilities. Second, we will incorporate AI-based image-processing systems such as automatic segmentation tools and combine these with our existing tools (e.g., allowing users to select an object that is then segmented from the entire video, and can be highlighted for comparison purposes). Third, we plan to expand the capabilities of our system to allow much larger numbers of videos at once: as seen in a few installations [39], large arrays of videos can be used almost as if they were pixels, allowing new visual patterns and features to emerge. Fourth, we will incorporate live video capabilities into our system, to allow us to develop techniques for real-time movement and gesture guidance using overlays (e.g., the YouMove system [2]). Fifth, we will test our techniques with other types of linear data, such as event streams and time-series data [20], which may also benefit from techniques such as small multiples, overlays, and translations between timeline views and temporal views. Sixth, we will conduct in-depth analyses of our more novel techniques (such as shadow markers) and conduct formal evaluations.

8 CONCLUSION

Comparison is a common task in visual analytics, but despite the wide variety of research into video manipulation and exploration, there is still little support for comparison tasks involving video. There are many scenarios in which users need to carry out video comparison, but current tools do not support these tasks; as a result, comparing even two videos typically requires several ad-hoc and tedious manual operations. To help address this problem, we developed a suite of interaction techniques that explicitly support video comparison. We provide techniques for equalizing the reference frame of videos to be compared, juxtaposition techniques for enhancing side-by-side and small-multiples comparisons, superposition techniques for comparing overlaid videos, explicit-encoding techniques that visualize differences between extracted points, and temporal-to-linear techniques that translate between a temporal sequence of frames and a 1D timeline. We incorporated all of our techniques into a web-based demonstration system with five different datasets, and evaluated the techniques with an analysis of steps required for basic tasks, as well as a user study that demonstrated the techniques' learnability, usability, and utility. Our work provides designers with new information about how to improve the task of comparing video: we identify core tasks for this domain, demonstrate several new techniques that specifically support different aspects of the tasks, and show that the techniques can be successfully understood and used by non-specialist users.

ACKNOWLEDGMENTS

REFERENCES

[1] Robert Amar, James Eagan, and John Stasko. 2005. Low-level components of analytic activity in information visualization. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.* IEEE, 111–117.

[2] Fraser Anderson, Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2013. YouMove: enhancing movement training with an augmented reality mirror. In *Proceedings of the 26th annual ACM symposium on User interface software and technology.* 311–320.

[3] Olivier Bau and Wendy E Mackay. 2008. OctoPocus: a dynamic guide for learning gesture-based command sets. In *Proceedings of the 21st annual ACM symposium on User interface software and technology.* 37–46.

[4] Fabian Beck, Michael Burch, Corinna Vehlow, Stephan Diehl, and Daniel Weiskopf. 2012. Rapid serial visual presentation in dynamic graph visualization. In *2012 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC).* IEEE, 185–192.

[5] Richard A Becker, William S Cleveland, and Ming-Jen Shyu. 1996. The visual design and control of trellis display. *Journal of computational and Graphical Statistics* 5, 2 (1996), 123–155.

[6] Jacques Bertin. 1981. *Graphics and graphic information processing.* de Gruyter.

[7] Sebastian Bremm, Tatiana von Landesberger, Martin Heß, Tobias Schreck, Philipp Weil, and Kay Hamacherk. 2011. Interactive visual comparison of multiple trees. In *2011 IEEE Conference on Visual Analytics Science and Technology (VAST).* IEEE, 31–40.

[8] Min Chen, Ralf Botchen, Rudy Hashim, Daniel Weiskopf, Thomas Ertl, and Ian Thornton. 2006. Visual signatures in video visualization. *IEEE Transactions on Visualization and Computer Graphics* 12, 5 (2006), 1093–1100.

[9] Nihanth W Cherukuru and Tim Scheitlin. 2020. Visual Comparator: An Interactive Tool for Dynamic Spatiotemporal Comparative Visualizations. *Bulletin of the American Meteorological Society* 101, 10 (2020), E1861–E1869.

[10] Ting-Hsuan Chou and Wei-Ta Chu. 2021. Automatic baseball pitch overlay. In *Proceedings of the 2021 International Conference on Multimedia Retrieval.* 599–602.

[11] Bruna CR Cunha, Diogo Pedrosa, Rudinei Goularte, and Maria da Graça Campos Pimentel. 2012. Video annotation and navigation on mobile devices. In *Proceedings of the 18th Brazilian symposium on Multimedia and the web.* 261–264.

[12] Fabian Lorenzo Dayrit, Ryosuke Kimura, Yuta Nakashima, Ambrosio Blanco, Hiroshi Kawasaki, Katsushi Ikeuchi, Tomokazu Sato, and Naokazu Yokoya. 2017. Remagicmirror: Action learning using human reenactment with the mirror metaphor. In *MultiMedia Modeling: 23rd International Conference, MMM 2017, Reykjavik, Iceland, January 4-6, 2017, Proceedings, Part I 23.* Springer, 303–315.

[13] William Delamare, Teng Han, and Pourang Irani. 2017. Designing a gaze gesture guiding system. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services.* 1–13.

[14] Pierre Dragicevic, Gonzalo Ramos, Jacobo Bibliowitcz, Derek Nowrouzezahrai, Ravin Balakrishnan, and Karan Singh. 2008. Video browsing by direct manipulation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* 237–246.

[15] Geoffrey Ellis and Alan Dix. 2007. A taxonomy of clutter reduction for information visualisation. *IEEE transactions on visualization and computer graphics* 13, 6 (2007), 1216–1223.

[16] Dustin Freeman, Hrvoje Benko, Meredith Ringel Morris, and Daniel Wigdor. 2009. ShadowGuides: visualizations for in-situ learning of multi-touch and whole-hand gestures. In *Proceedings of the ACM international conference on interactive tabletops and surfaces.* 165–172.

[17] Emilien Ghomi, Stéphane Huot, Olivier Bau, Michel Beaudouin-Lafon, and Wendy Mackay. 2013. Arpège: Learning Multitouch Chord Gestures Vocabularies. *ITS 2013 - Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces,* 209–218. https://doi.org/10.1145/2512349.2512795

[18] Michael Gleicher. 2017. Considerations for visualizing comparison. *IEEE transactions on visualization and computer graphics* 24, 1 (2017), 413–423.

[19] Michael Gleicher, Danielle Albers, Rick Walker, Ilir Jusufi, Charles D Hansen, and Jonathan C Roberts. 2011. Visual comparison for information visualization. *Information Visualization* 10, 4 (2011), 289–309.

[20] Tovi Grossman, Justin Matejka, and George Fitzmaurice. 2010. Chronicle: capture, exploration, and playback of document workflow histories. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology.* 143–152.

[21] Carl Gutwin, Michael van der Kamp, Md Sami Uddin, Kevin Stanley, Ian Stavness, and Sally Vail. 2019. Improving early navigation in time-lapse video with spread-frame loading. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems.* 1–12.

[22] Hans Hinterberger. 2010. The VisuLab: An Instrument for Interactive, Comparative Visualization. *Technical Report/ETH Zurich, Department of Computer Science* 682 (2010).

[23] Markus Hoeferlin, Benjamin Hoeferlin, Gunther Heidemann, and Daniel Weiskopf. 2013. Interactive schematic summaries for faceted exploration of surveillance video. *IEEE Transactions on Multimedia* 15, 4 (2013), 908–920.

[24] Cristian Hofmann, Uwe Boettcher, and Dieter W Fellner. 2010. Change awareness for collaborative video annotation. In *Proceedings of COOP 2010: Proceedings of the 9th International Conference on Designing Cooperative Systems, May, 18-21, 2010, Aix-en-Provence.* Springer, 101–117.

[25] Dan Jackson, James Nicholson, Gerrit Stoeckigt, Rebecca Wrobel, Anja Thieme, and Patrick Olivier. 2013. Panopticon: A parallel video overview system. In *proceedings of the 26th annual ACM symposium on User interface software and technology.* 123–130.

[26] David E Jacobs, Dan B Goldman, and Eli Shechtman. 2010. Cosaliency: Where people look when comparing images. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology.* 219–228.

[27] Waqas Javed and Niklas Elmqvist. 2012. Exploring the design space of composite visualization. In *2012 ieee pacific visualization symposium.* IEEE, 1–8.

[28] Waqas Javed, Bryan McDonnel, and Niklas Elmqvist. 2010. Graphical perception of multiple time series. *IEEE transactions on visualization and computer graphics* 16, 6 (2010), 927–934.

[29] Sasa Junuzovic, Kori Inkpen, Rajesh Hegde, Zhengyou Zhang, John Tang, and Christopher Brooks. 2011. What did I miss? In-meeting review using multi-modal accelerated instant replay (AIR) conferencing. In *Proceedings of the sigchi conference on human factors in computing systems.* 513–522.

[30] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. *arXiv preprint arXiv:2304.02643* (2023).

[31] Jiazhou Liu, Arnaud Prouzeau, Barrett Ens, and Tim Dwyer. 2020. Design and evaluation of interactive small multiples data visualisation in immersive spaces. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR).* IEEE, 588–597.

[32] María-Jesús Lobo, Emmanuel Pietriga, and Caroline Appert. 2015. An evaluation of interactive map comparison techniques. In *Proceedings of the 33rd annual ACM conference on human factors in computing systems.* 3573–3582.

[33] Sehi LYi, Jaemin Jo, and Jinwook Seo. 2020. Comparative layouts revisited: Design space, guidelines, and future directions. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2020), 1525–1535.

[34] Justin Matejka, Michael Glueck, Erin Bradner, Ali Hashemi, Tovi Grossman, and George Fitzmaurice. 2018. Dream lens: Exploration and visualization of large-scale generative design datasets. In *Proceedings of the 2018 CHI conference on human factors in computing systems.* 1–12.

[35] Justin Matejka, Tovi Grossman, and George Fitzmaurice. 2012. Swift: reducing the effects of latency in online video scrubbing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* 637–646.

[36] Justin Matejka, Tovi Grossman, and George Fitzmaurice. 2013. Swifter: improved online video scrubbing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* 1159–1168.

[37] Justin Matejka, Tovi Grossman, and George Fitzmaurice. 2014. Video lens: Rapid playback and exploration of large video collections and associated metadata. In *Proceedings of the 27th annual ACM symposium on User interface software and technology.* 541–550.

[38] Amir H Meghdadi and Pourang Irani. 2013. Interactive exploration of surveillance video through action shot summarization and trajectory visualization. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2119–2128.

[39] Ken Murphy. 2011. A History of the Sky. http://www.murphlab.com/2011/11/15/a-history-of-the-sky-for-one-year/. Accessed: 2023-09-14.

[40] Christine M Neuwirth, Ravinder Chandhok, David S Kaufer, Paul Erion, James Morris, and Dale Miller. 1992. Flexible diff-ing in a collaborative writing system. In *Proceedings of the 1992 ACM conference on Computer-supported cooperative work.* 147–154.

[41] Christina Niederer, Holger Stitz, Reem Hourieh, Florian Grassinger, Wolfgang Aigner, and Marc Streit. 2017. TACO: visualizing changes in tables over time. *IEEE transactions on visualization and computer graphics* 24, 1 (2017), 677–686.

[42] Vanessa Peña-Araya, Anastasia Bezerianos, and Emmanuel Pietriga. 2020. A comparison of geographical propagation visualizations. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems.* 1–14.

[43] Ghulam Jilani Quadri and Paul Rosen. 2021. A survey of perception-based visualization studies by task. *IEEE transactions on visualization and computer graphics* (2021).

[44] Gustavo Rovelo, Donald Degraen, Davy Vanacken, Kris Luyten, and Karin Coninx. 2015. Gestu-wan-an intelligible mid-air gesture guidance system for walk-up-and-use displays. In *Human-Computer Interaction–INTERACT 2015: 15th IFIP TC 13 International Conference, Bamberg, Germany, September 14-18, 2015, Proceedings, Part II 15.* Springer, 368–386.

[45] Arne Schipper, Hauke Fuhrmann, and Reinhard von Hanxleden. 2009. Visual comparison of graphical models. In *2009 14th IEEE International Conference on*

Engineering of Complex Computer Systems. IEEE, 335–340.

[46] Klaus Schoeffmann, Marco A Hudelist, and Jochen Huber. 2015. Video interaction tools: A survey of recent work. *ACM Computing Surveys (CSUR)* 48, 1 (2015), 1–34.

[47] Hyunjoo Song, Bongshin Lee, Bo Hyoung Kim, and Jinwook Seo. 2012. DiffMatrix: Matrix-based Interactive Visualization for Comparing Temporal Trends.. In *EuroVis (Short Papers)*.

[48] Justin Talbot, Vidya Setlur, and Anushka Anand. 2014. Four experiments on the perception of bar charts. *IEEE transactions on visualization and computer graphics* 20, 12 (2014), 2152–2160.

[49] James Tam and Saul Greenberg. 2004. A framework for asynchronous change awareness in collaboratively-constructed documents. In *Groupware: Design, Implementation, and Use: 10th International Workshop, CRIWG 2004, San Carlos, Costa Rica, September 5-9, 2004. Proceedings 10*. Springer, 67–83.

[50] John C Tang, Gina Venolia, and Kori M Inkpen. 2016. Meerkat and periscope: I stream, you stream, apps stream for live streams. In *Proceedings of the 2016 CHI conference on human factors in computing systems*. 4770–4780.

[51] Christian Tominski. 2016. Comparing: reducing costs of visual comparison. In *Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: Short*

Papers. 137–141.

[52] Christian Tominski, Camilla Forsell, and Jimmy Johansson. 2012. Interaction support for visual comparison inspired by natural behavior. *IEEE Transactions on visualization and computer graphics* 18, 12 (2012), 2719–2728.

[53] Milka Trajkova and Francesco Cafaro. 2016. E-ballet: designing for remote ballet learning. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. 213–216.

[54] Hideaki Uchiyama, Shunsuke Sakurai, Masashi Mishima, Daisaku Arita, Takashi Okayasu, Atsushi Shimada, and Rin-ichiro Taniguchi. 2017. An easy-to-setup 3D phenotyping platform for KOMATSUNA dataset. In *Proceedings of the IEEE international conference on computer vision workshops*. 2038–2045.

[55] Md Sami Uddin, Carl Gutwin, and Andy Cockburn. 2017. The effects of artificial landmarks on learning and performance in spatial-memory interfaces. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 3843–3855.

[56] Romain Vuillemot. 2011. Watching +100 The Simpsons episodes at the same time (experiment). https://www.youtube.com/watch?v=Fg8Ej8P8dcw. Accessed: 2023-09-14.