DOI: 10.1111/cgf.13763

COMPUTER GRAPHICS *forum*
Volume 39 (2020), number 1 pp. 231–243

Memento: Localized Time-Warping for Spatio-Temporal Selection

V. Solteszova[1,2], N. N. Smit[1,3], S. Stoppel[1], R. Grüner[3,4] and S. Bruckner[1]

[1]Department of Informatics, University of Bergen, Bergen, Norway
{veronika.solteszova, noeska.smit, sergej.stoppel, stefan.bruckner}@uib.no
[2]Technology Department, NORCE – Norwegian Research Centre AS, Bergen, Norway
[3]Mohn Medical Imaging and Visualization Centre, Department of Radiology, Haukeland University Hospital, Bergen, Norway
Renate.Gruner@uib.no
[4]Department of Physics and Technology, University of Bergen, Bergen, Norway

Abstract

Interaction techniques for temporal data are often focused on affecting the spatial aspects of the data, for instance through the use of transfer functions, camera navigation or clipping planes. However, the temporal aspect of the data interaction is often neglected. The temporal component is either visualized as individual time steps, an animation or a static summary over the temporal domain. When dealing with streaming data, these techniques are unable to cope with the task of re-viewing an interesting local spatio-temporal event, while continuing to observe the rest of the feed. We propose a novel technique that allows users to interactively specify areas of interest in the spatio-temporal domain. By employing a time-warp function, we are able to slow down time, freeze time or even travel back in time, around spatio-temporal events of interest. The combination of such a (pre-defined) time-warp function and brushing directly in the data to select regions of interest allows for a detailed review of temporally and spatially localized events, while maintaining an overview of the global spatio-temporal data. We demonstrate the utility of our technique with several usage scenarios.

Keywords: interaction, temporal data, visualization, spatio-temporal projection

ACM CCS: • Human-centred computing → Visualization techniques, Scientific visualization, • Mathematics of computing → Time series analysis

1. Introduction

Spatio-temporal data are defined as data that represent spatial structures that dynamically change over time. This type of data is widely acquired and analysed in a variety of domains, such as climatology, medicine, physics and environmental science [AKK18]. Analytical tasks in spatio-temporal visualization include studying changes over time, comparing different points in space and time and uncovering spatio-temporal patterns [KH13]. While spatio-temporal datasets are ubiquitous, they are notoriously hard to visualize particularly when there is more than one dimension in addition to time [BDA*17]. One way of dealing with the complexity of the data is to abstract the temporal component of the data by computing static summaries (aggregates) [AAG03].

When summarization via aggregates is not suitable for the task at hand, interaction can be employed to allow users to inspect spatio-temporal events of interest. Interaction techniques for time-dependent spatial data often focus on affecting the spatial aspects of the data, for instance the manipulation of camera or clipping planes, or adjustment of the transfer function to alter the visual representation of the spatial data characteristics. The temporal aspect of data interaction is often neglected, and the spatio-temporal data are then visualized as individual time steps, an animation over the time steps or a static summary (aggregation) over the temporal domain. These techniques are unable to cope with the task of re-viewing an interesting local spatio-temporal event, while continuing to observe the rest of the feed. For streaming data in particular, rewinding and pausing amounts to missing information from the current live feed.

In this paper, we present a spatio-temporal data interaction technique that enables users to specify spatio-temporal areas of interest, and manipulate temporal and visual aspects of these regions locally (as demonstrated in Figure 1). Users can interactively specify spatial regions of interest, whose temporal behaviour is altered by a time-warp function, enabling the investigation of events while still maintaining a global overview. Our approach is particularly suitable in streaming data visualization scenarios. The main contributions of this paper can be summarized as follows:

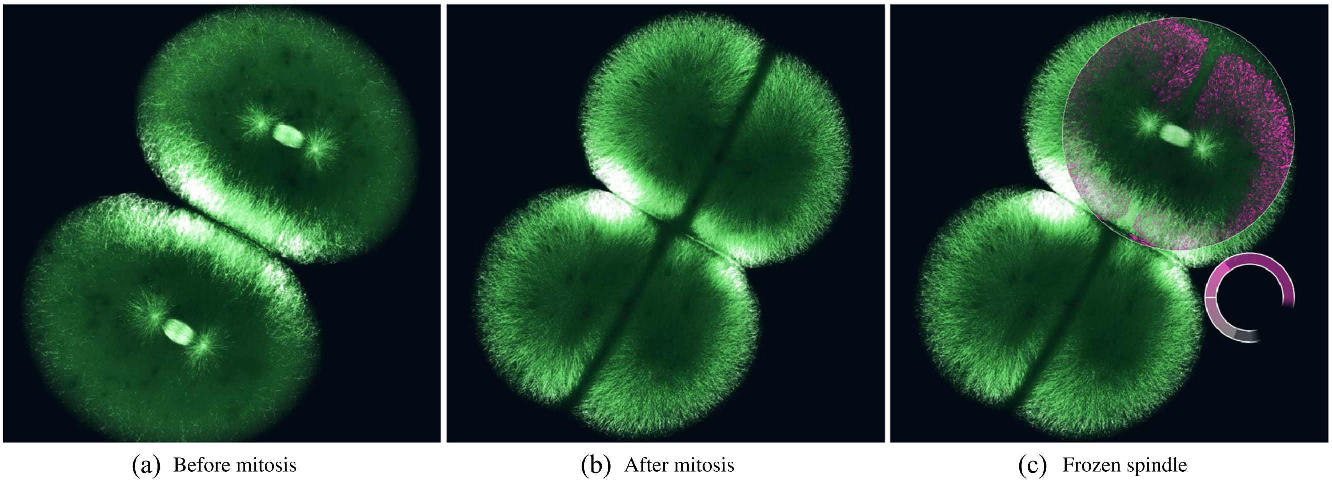


Figure 1: *In the live-cell imaging usage scenario, we can observe a mitotic spindle at the centre of the cells before mitosis occurs (a). After mitosis, the spindle is no longer visible (b). Using a time-warping function, we are able to travel back in time to freeze the spindle for closer observation locally, while the rest of the dynamics continues as normal (c).*

- We propose a spatio-temporal data interaction technique that allows for local temporal manipulation of a focus region, while the context region remains observable.
- We present the concept of time-warp functions, which allows for local time manipulation via template or custom functions.
- We demonstrate the potential of our localized time-warping technique in several usage scenarios from a variety of domains.

2. Related Work

Efforts in capturing the temporal dimension in nature have been made before the birth of visualization as a scientific discipline. Examples include photography or drawings illustrating movement over time. In photography, this technique is called multiple exposures, where a moving object appears several times over a still background. Mili [Mil] used this technique to effectively communicate the trajectory of objects, or the order of movements in dance or sport exercises. In comic drawings, techniques that depict the temporal dimension include motion lines, and repetition of object outlines at different timepoints [McC93].

Today, there is a large body of related work in the field of spatio-temporal data visualization, and there are several comprehensive surveys that focus on this topic. Adrienko *et al.* [AAG03] presented an analytical review of exploratory spatio-temporal geovisualization, focusing mainly on the types of data and exploratory tasks that are supported. Aigner *et al.* [AMM*07] provided a systematic view of time-oriented data and discussed various techniques for visualizing time. They proposed a classification scheme, based on time, data and representation characteristics. The survey by Kehrer and Hauser [KH13] describes visualization and visual analysis of multifaceted scientific data. Here, one of the facets discussed is spatio-temporal data visualization. To frame our contribution in the context of related scientific work, we have reviewed different approaches in information visualization, visualization of scientific spatial data, as well as video visualization. Here, we list only the most closely related prior works and categorize them into static versus dynamic representations of spatio-temporal data.

Static representations In this section, we highlight several approaches that provide a static overview of the temporal data component as a summary. Bach *et al.* [BSH*16] introduced time functions, a time folding approach to visualize temporal evolution patterns in data. A timeline visualization is folded into itself in such a way that similar time points are close to each other. In the example case study based on collaborative text editing, their approach allowed to see how fast the text was edited, or if two authors were concurrently editing the same text. Individual time steps are visualized as points, whose mutual distance relates to quantifiable measures of similarity between corresponding time frames. Inspired by chronophotography, a technique capturing movement in several frames, Woodring and Shen [WS03] employed time integration to communicate several time steps of time-varying volumetric datasets at once. The technique provided the possibility to analyse a time-varying dataset as a whole, instead of analysing frames individually. Balabanian *et al.* [BVMG08] introduced interaction mechanisms for communicating temporal characteristics of time-varying volumes through temporal style transfer functions. The goal was to convey features from multiple time steps in a single image. They employed temporal projections, such as maximal temporal intensity, maximum temporal average, and maximum density change. Their transfer function concept assumed that all time steps are provided, and as such is not suitable for streaming data. Frey and Ertl [FE17] recently presented an approach to select time steps from time-dependent volumetric data adaptively, in order to compose an integrated visualization of spatial structure and temporal development in one combined rendering. A minimum cost flow-based technique optimized the selection of the time steps by determining distances between time steps. In a follow-up paper by Frey [Fre18], this approach was extended with a technique to extract spatio-temporal contours using deep volume raycasting. The generated contours depict where and when non-continuous changes occur or spatial bounds are present in a single summary overview, using colour to encode temporal information. Other recent work by Zhou and Chiang [ZC18] proposed a key time step selection mechanism for time-varying volumetric datasets, based on information theoretical measures. Recent work by Schmidt *et al.* [SFP*18] presented Popup-plots, a technique to warp temporal charts using 3D rotation. In this case, 3D datasets containing two dependent variables and time as an independent variable can be inspected by transforming space using an ellipsoidal model.

In contrast to our approach, these methods aim to produce a single static visual overview of a temporal dataset, while our goal is to spatially modify time locally in order to better understand specific events in time-series or streaming data.

Dynamic representations In this section, we provide a brief overview of methods that focus on improving the dynamic display of spatio-temporal data, which is also the category that our work falls into. Wang *et al.* [WYM08] presented an importance-based approach for the visual analysis of data with a temporal component. They used the term *block* to refer to a local spatial neighbourhood around an important feature. They presented an algorithm that suggests key time steps from a long sequence of data, given a particular importance metric. In their work, Höferlin *et al.* [HHWH11] presented an adaptive fast-forward approach, which determines an optimal playback speed, based on the approximated information content in the video, in order to obtain compact surveillance video streams. Le Muzic *et al.* [LMWPV15] introduced illustrative time lapse, an automatic approach for the creation of illustrative animations based on a simulation in cell biology. Their method includes seamless temporal zooming, visual abstraction and simplification of molecular trajectories, enhanced visual focus on relevant events and lens effects to keep a realistic representation of the environment in the context. Seamless temporal zooming amounts to interactively changing the temporal resolution of the generated illustrative animation. The lens effect means that the velocity of the particles in the simulation is locally affected based on the zoom level. Animation generation can be thought of as a data production process, while, in contrast to this, we aim at the visualization of existing data. We would also categorize this approach as object based, because the scene is modified by changing the objects (their location and speed), while in our approach we modify only the visual representation.

Techniques for time-dependent data visualization can be further categorized into two groups: analysis and process monitoring techniques [Sme14]. Analysis techniques are generally executed after data acquisition, for example on a dataset that is extracted from a data stream. In contrast, process monitoring is targeted at *in-situ* exploration of parameters and results. The reason is that analytic tasks require more time and the operator would lose the overview of the running stream while performing the analysis. Furthermore, in many scenarios, the entire data stream cannot be recorded for analysis. Even though efforts have been made to overcome the storage limitations for certain data types [LH11], this still remains a challenge in general. Smestad's thesis [Sme14] focuses on information visualization without the spatial aspect, but some of the concepts discussed, such as *decision pressure* and *conflict of attention*, apply to spatial data visualization as well.

In addition to time-varying data visualization, our method is also related to work dealing with focus-and-context visualization, since we warp time in certain regions of interest, or focus regions, in order to give them more prominence than others. Löffelmann and Gröller [LG96] introduced extended cameras, where one possible application is the generation of local non-linear zoom effects. Regions of interest (focus) can be thus given more space in the visualization than the context. Mindek *et al.* [MBG13] presented contextual snapshots, a concept that combines spatial selections with their global context, and proposed an effective management technique for the selected data in subsequent work [MGB14]. Their approach focuses on static spatial data and provides contextual snapshots for single time steps. In our approach, rather than focusing on pure spatial snapshots, we further extend this concept to a spatio-temporal snapshot.

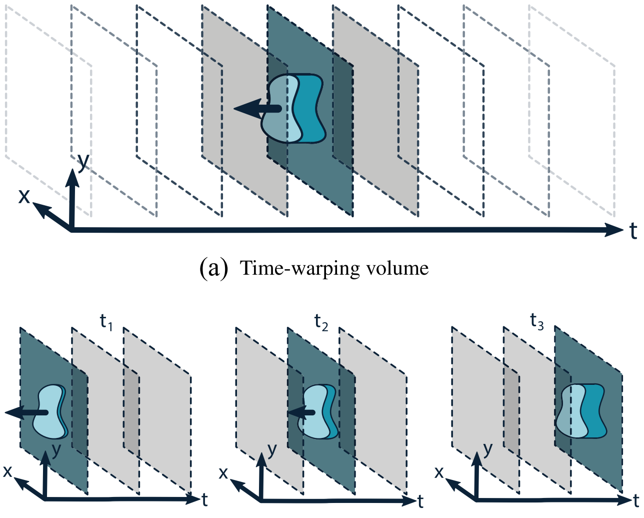
We have also reviewed how the concept of time warping is used in fields other than visualization, such as multimedia and signal processing, including video and image processing. Dynamic time warping is a well-established algorithm in signal processing for quantifying signal similarities, frequently applied, e.g. in speech recognition or event detection in videos [Mül07] and image stabilization in videos [BBB*08]. For presentation purposes, video playback can also be locally manipulated in some commercial video editing tools, e.g. Adobe Premiere Pro, by using masking and playback speed adjustments. However, there are several effects that either cannot be achieved as easily as in our solution, or not achieved at all. First, our concept is applicable to any kind of visualization of temporal data, including streaming data, regardless of its dimensionality. That means our approach is applicable not only to video data, but also to interactive visualization of volumetric data as well as non-spatial information visualization. Second, our approach allows for easy adjustment of the local time-warping function while ensuring global time consistency after the inspection, as opposed to direct manipulation of local playback speed. In our approach, playback speed is indirectly manipulated as we adjust the time offset, and reset the offset to zero as the manipulation ends. This effect of local time-warping and catching up would be difficult to achieve by playback speed manipulation.

Bach *et al.* [BDA*17] presented a descriptive framework for temporal data visualization based on generalized space-time cubes. In this model, visualizations can be described as combinations of elementary operations on this cube. In the next section, we will describe our visualization method referring to these operations where possible.

3. Time-Warping Concept

Focus-and-context visualization in information visualization and in spatial visualization of 2D and 3D datasets has become a well-established method for exploration and analysis of data. However, this approach has been less explored for data with a temporal component.

The first motivation for a focus-and-context technique for time-varying data comes from scenarios where the *decision pressure* is high, i.e. the operator is required to execute fast, but informed decisions. Pausing and rewinding the data stream might not be desired due to *conflict of attention* – the data stream is running and meanwhile other important events can escape the operators' attention.



(a) Time-warping volume

(b) Temporal deformation changes over time

Figure 2: *Our time-warping operation can be conceptually seen as a deformation of a spatial selection in an image towards other time frames, within a sliding temporal window, that is indicated by grey frames (a). As time progresses, the deformation of the time-warp volume may change (b). We indicate the deformation direction with the arrows on the time-warp volume.*

Instead, they need to warp the time locally around the occurrence of a certain event. For example, this concerns commentators of live video streams, operators in the oil and gas industry where delays come at great cost or scenarios involving large-scale simulations where it is possible to adjust parameters, but the entire simulation cannot be stored and revisited, so the adjustments need to happen *in-situ*. A second motivation to develop a focus-and-context technique for temporal data comes from the need for the comparison of events that have different spatial and temporal locations in the dataset. For example, in neuroimaging and cognitive and behavioural neuroscience, researchers analyse neuronal networks in the brain, their function and the interplay between individual networks. Events that happen in different networks might be similar, but delayed in time and have different spatial locations. However, such hypotheses are hard to verify with current visualization tools that are limited to juxtaposed viewing and numerical analysis.

We propose a method to alter the playback locally in a region of interest, while maintaining an overview of the rest of the feed. This method is based on the concept of specifying a time-warping volume within the space-time cube (Figure 2a), which allows the user to perform spatio-temporal projection operations within the selection via compositing operations. The time-warping volume consists of a deformation in the temporal direction of the space-time cube for a given spatial selection. The deformation may change over time, within a sliding temporal window (Figure 2b). Given this time-warping volume, several compositing operations may be performed to visualize local behaviour over time.

The two required main ingredients to implement our concept in practice are the specification of the time-warping volume and its behaviour over time, and a visual encoding that is suitable for displaying the resulting time-warp over time. However, the simultaneous specification of both the spatial and temporal aspects of the time-warping volume is difficult. Therefore, we subdivide this specification into two separate components. Our time-warping method then consists of three major components:

- A specification of what the temporal behaviour of the deformation over time should be.
- A specification of the spatial selection, which should be affected by the time-warp.
- A visualization of the spatio-temporal selection.

While our concept allows for many different specification and visualization methods, in this paper, we are inspired by how transfer functions assign colour and opacity to intensity values. Therefore, we allow users to design a time-warping function to control the temporal behaviour. We then perform a spatio-temporal projection, whose behaviour is determined by this time-warping function. We choose to employ a straight-forward brushing operation for the spatial selection, and design our visualization as a compositing operation within the time-warping volume.

3.1. Spatio-temporal projection

In contrast to straight-forward playback, our time-warping concept is not restricted to showing one single time step at a time. In fact, several time steps can be combined into the same 2D visualization at once, by projecting multiple 3D datasets into 2D. In our case, the projection is applied to selected regions of interest, which can be defined by the user for instance via brushing operations. In a final composition step, the output of the spatio-temporal projection within the selected region and the original time-varying data at the current timestep are combined into one visualization for display to the user.

Projection is, generally speaking, a mapping from a higher dimensional space to a lower dimensional space. In our case, we have n spatial dimensions and a temporal dimension, and we project the data into a visualization with two spatial dimensions and one temporal dimension. For example, in the case of streaming volumetric data, we project from three spatial dimensions and time, and in the case of video data, we have two spatial dimensions and time.

The spatial domain of the mapping is user-defined via a region of interest, and in the temporal domain with a starting time step t_0, as well as a duration, called the time-warp window W_t. In our prototype, W_t is a preset that can be changed manually. Spatio-temporal projection then maps from space-time to 2D space, such that the co-domain of the mapping is a 2D image.

The mapping function, f_t, defines the relation between the time step that is shown, and time step that corresponds to the global current time step of the stream. For generality, we normalize the input and output value ranges of the mapping function f to the unit intervals. The final *time-warping function* can then be defined as:

$$t' = W_t f_t \left(\frac{t - t_0}{W_t} \right) + t_0. \tag{1}$$

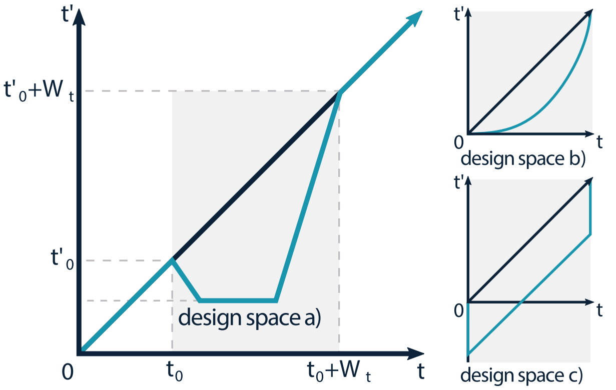


Figure 3: *Explanation of the time-warp (blue line) with respect to the identity mapping (black line). The interval* $[t_0, t_0 + W_t]$ *defines the design space for the time-warp function. Design (a) shows a setup, where the time flow is reversed, frozen and then catches up. Design (b) shows a slow down of the flow. Design (c) shows a delay, with playback at the same speed as the original. Note that the time-warping function is parallel to the identity function.*

We illustrate the effects of three time-warping functions in Figure 3. Outside of the time interval $[t_0, t_0 + W_t]$, the function is the identity function, and is not subject to any specific design possibilities. For a straight-forward video viewing example, the time-warping function coincides fully with the identity function $t' = t$.

In Figure 3, we illustrate several different potential time-warping function designs. For freezing the feed, the function is constant $t' = t_0$ in some part of the time window (see Figure 3: left). For slowing down time and then catching up, the slope should be first less steep than the identity function (see Figure 3: top right). To reverse the flow of time, the function has to have a negative slope (see Figure 3: bottom right). We impose some practical constraints on the design of these functions in order to have a smooth transition from normal time to warped time. The first condition is that at time t_0 and $(t_0 + W_t)$, the function should intersect the identity function. This means that the start of the time-warp will be identical to the global time, and that after the time-warp is over, the warped time catches up with global time once again. Furthermore, we do not allow $t' > t$, since in case we are dealing with data from a feed, we cannot travel to future frames. For this reason, any parts of the function above the identity function will be clamped to the identity function.

The time-warping functions relate to the space-time cube concept, visible in Figure 2, as follows. The time-warping function is responsible for determining how the deformation of the time-warp volume behaves over time. In Figure 2(b), we see the beginning of a deformation that results in the spatial selection travelling back in time over time (indicated by the black arrows in the figure), meaning that for every frame within the time-warp window, the deformation extends further along the negative time axis, and thus the selection travels further back in time. Such a deformation may be achieved by simply specifying a time-warping function with a negative incline in some parts of the time-warp window.

In principle, many more types of time-warp volumes are possible than what we have constrained our time-warping functions to represent. For instance, a free-form deformation in the temporal axis of the time-warp volume would lead to warping to more than one timestep within the spatio-temporal selection over time. Taking the interaction time into account while specifying the spatio-temporal region of interest would lead to similarly complex time-warping operations. While such deformations are certainly possible conceptually, in our experiments we found them to be rather difficult to interpret. Thus, we mainly focus on time-warping function designs that correspond to time-warping actions that are easier to interpret, such as slowing down time, delaying time, freezing time, speeding up time and combinations of these actions.

3.2. Spatio-temporal selection

Selection tools for time-varying data must be effective and allow for quick selection of spatial regions. Depending on the data, there might be different requirements on the geometry of the selection. In our approach, we provide the following selection types:

- **Additive circular brush.** The user creates a selection by brushing with a circle of adjustable radius over the visualization. All 'painted' pixels become part of the selection. The radius of the brush can be increased or decreased using keyboard shortcuts or via the graphical user interface.
- **Lasso tool.** The user draws an outline of the selection and the entire enclosed region becomes part of the selection.
- **Non-additive circular brush.** The user picks the centre of the selection and drags the mouse to define the radius. The surface area of the circle will define the selection.
- **Non-additive rectangular brush.** The user picks one corner of the rectangle and drags the mouse to define the opposite corner of the rectangle.

Even though in most cases we used the additive circular brush for its quick response time, other types of brushes could be useful for selecting similarly shaped geometry, or for those selections that need to follow a certain shape type.

The selection is represented as a 2D mask, where each pixel encodes the offset of the time-warping function from the identity function:

- In our spatio-temporal projection examples, we warp the entire time-warp volume into one single time step in the past. Therefore, the 2D selection mask is binary $t' \in \{0, 1\}$ for context and selected pixels, respectively. The offset between the identity function and the time-warping function is determined for the entire selection globally using the formula $\frac{t-t_0}{W_t}$.
- Our concept of warping allows us to go further and theoretically warp multiple time steps simultaneously. In such a case, the selection would not be binary, but it would encode $\frac{t-t_0}{W_t}$ with $t' \in [0, 1]$, where 0 is mapped to the selection on starting frame and 1 to the current frame.

3.3. Time-warping visualization

Given our spatio-temporal selection, and the definition of what should happen over time via the time-warping function design,

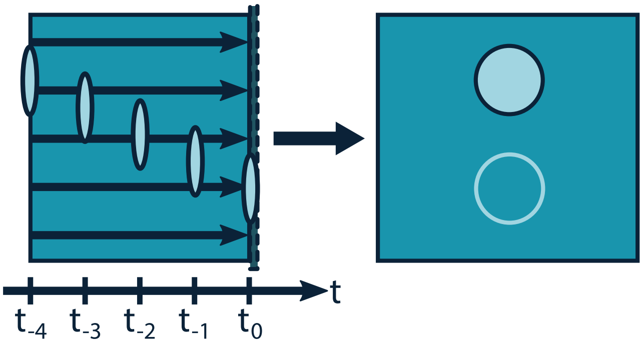


Figure 4: *An illustration of time-warp volume compositing: in our approach, we are currently only visualizing information from the current global timestep t_0 blended with information from the timestep furthest back in time, here at t_{-4} (left). In this illustration, this results in the moving circle being visible twice in the final image (right). Other compositing schemes, such as temporal maximum intensity projection over all timesteps, are also possible.*

there is also a need to visualize the resulting time-warp volume. Here, there are two main components that require an appropriate visual encoding: temporal compositing, e.g. visualization for the time-warped region of interest (parallel timeline) within its context, i.e. the identity mapping of the dataset, and a comprehensive visualization of time itself.

The first visualization challenge lies in the visualization of the time-warp volume, i.e. the parallel timeline, and the context, i.e. the global timeline or identity mapping. Both the parallel and global timelines are represented as a spatio-temporal projection onto a 2D image. The time-warp volume is spatially defined through the 2D selection and temporally through the starting frame and the time-warp window size W_t. The size W_t defines how many frames are buffered. For example, for video data, the 2D selection mask defines t' for each pixel of the rendered video. For streamed volumetric data, the 2D selection mask defines t' for each viewing ray, so the time-warp volume is projected using a raycaster, where each ray has its own t' that determines which time step this ray belongs to.

As we have two visualizations of two timelines, the time-warped image and the global time step image, and the goal is to compare them also spatially, there is a need to communicate information from both timelines at the same time. This amounts to a comparative visualization problem. In comparative visualization, techniques are typically categorized as juxtaposition (side by side), superposition (overlaying in the same space) and explicit encoding [GAW*11]. In our case, due to the temporal nature of the visualization, time spent on switching attention spatially is time lost observing the original visualization, and therefore juxtaposition is not desirable. This leaves us with superposition and explicit encoding as comparative visualization options. Both pose challenges if we want to avoid visual clutter and confusion for the user. We have considered several options for the visual encoding of both timelines in a single spatial location:

- **Checkerboard.** We could interleave two timelines using a checkerboard scheme. In this case, square-shaped parts of both timelines are hidden, but the choice of which parts are hidden is solely determined by the chess-board mask, and not by the importance of the features, making it less suitable for our purposes.
- **Blending colour schemes.** We could also represent both timelines in different colour schemes and blend the colours. However, choosing a generically applicable suitable colour map for both timelines can also be difficult and represent an interesting research question in and of itself.
- **Blending styles.** Rather than using different colours for blending, different rendering styles may also be employed, for instance by displaying the warped timestep in the original style and the current global timestep with a modified style that only depicts the most important features.

We utilize blending of styles in our approach, as it allows us to communicate the time-warped information fully, while giving a rough overview of the current global timestep, enabling comparison in a single view. Different styles can be applied to specific parts of the space-time cube, or the whole data via compositing operations. In Figure 4, we illustrate this concept. For example, the compositing operation may include information from all time steps in the time-warp volume, and create a summary of the events in these time steps using a temporal maximum intensity projection. In our current implementation, we have limited the options for representation to style blending of the current global timepoint and the time-warped timepoint furthest back in time, based on the current time-warp. Other potentially promising avenues of more complex compositing schemes may be pursued as future work.

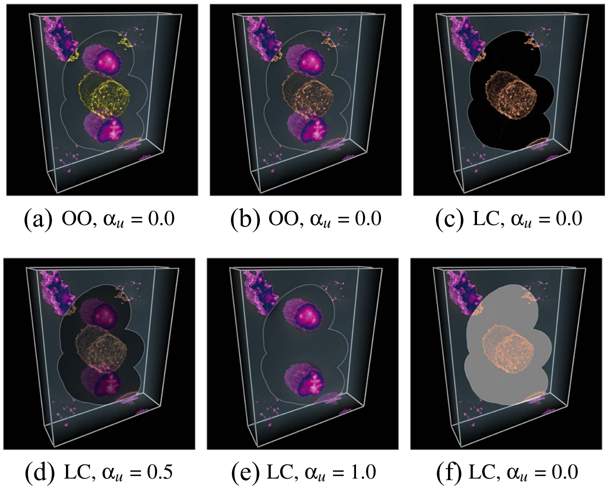
For the visual style of the time-warped region, the simplest way of visualizing the selection is only showing the time-warped information. Another naive approach is to simply use alpha blending on the original visual representations of both the current and time-warped timestep. However, alpha blending tends to result in a cluttered and washed out appearance, and it would be hard to visually determine which information comes from which timestep. To reduce the clutter while still allowing spatial comparison of the visible features, we employ an importance-driven approach. Based on an importance metric of choice, we construct a texture containing an importance value I per pixel and an additional blending parameter $\alpha_u \in [0, 1]$ set by the user. The blending parameter α_u modulates the opacity of the importance texture as $\alpha_i = I + \alpha_u$ and is then clamped to [0,1].

We refer to the time-warped image as $C_w\alpha_w$, where C_w are the RGB components of the time-warped image and α_w is its alpha component. Then, the final image C is composed using the over operator, i.e. alpha blending the importance texture over the time-warped image:

$$C = \frac{I\alpha_i + C_w\alpha_w(1-\alpha_i)}{\alpha_i + \alpha_w(1-\alpha_i)}. \tag{2}$$

This formulation allows the user to easily control the contribution of the rendered image during the time-warp, and the contours of the rendered image during the global time step.

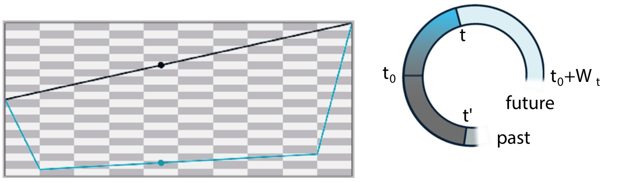
To saliently visualize the temporal differences, we choose to employ an edge detector to obtain an importance mask. In the first step, we compute the edges in the rendered image in the global time step



(a) OO, $\alpha_u = 0.0$ (b) OO, $\alpha_u = 0.0$ (c) LC, $\alpha_u = 0.0$

(d) LC, $\alpha_u = 0.5$ (e) LC, $\alpha_u = 1.0$ (f) LC, $\alpha_u = 0.0$

Figure 5: *Demonstration of possible composition schemes using alpha-blending (AB) and linear composition (LC) with varying α_u, and varying edge colour (EC) and background colour (BC). In (a), the EC is set to fully opaque yellow, (b)–(f) orange. In (a)–(e), the background colour is set to fully transparent, in (f) to fully opaque grey. This visualization was generated by rendering a 4D series from the Cell Image Library (CIL:13368) [PMK*].*



(a) Time-warping curve design widget (b) The warpometer widget

Figure 6: *The time-warping function design widget (a) features frames as units on both the* x- *and* y*-axis. The black dot indicates the current position on the identity function within the time window. The blue dot indicates the current warped time on the time-warping function. The distance between the black and blue dot signifies the current temporal offset. The warpometer widget (b) shows the current time t, warped time t' in relation to the time window $t_0 + W_t$.*

(identity mapping) using the Sobel edge detection filter to get value of I for each pixel. This formulation then emphasizes the edges as important features, while defining the remaining region ($I \sim 0$) as *background*. The overall appearance of the selection can be modulated with an additional parameter α_u, the colour of the edges (EC) and the background colour (BC). Figure 5 illustrates different alpha parameters and colours and the over operator (OO) compared to linear combination using α_u (LC) of the time-warped and the global time step rendering.

This Sobel filter-based edge enhancement is only one example of extracting descriptive features from an image that could be used within this context. Other options would include displaying apparent ridges, suggestive contours, ridges and valleys or contours only [JDA07].

An effective visualization of the time progression is a challenging task. Orientation in the temporal domain is not easy based on viewing the projected information alone, especially when the temporal dimensions are warped. Therefore, we designed two widgets, which both serve different purposes. First, we designed an inset timeline visualization that depicts the time-warp curve (Figure 6a). The black graph illustrates the normal timeline (based on the identity time function), and the blue graph consists of the time-warping function. The black dot symbolizes the temporal location of the context outside of the region of interest (and edge enhancement within if selected), on the normal timeline. The blue dot symbolizes the temporal location of the region of interest. This widget communicates the time-warping function design and can be switched off if desired. The rectangle grid in the background symbolizes frame or time units. In this example, the time window has 10 frames (time units). During the first time unit after selection (t_0), the warped data stream rewinds nine data frames back. Then, for the next eight time units, it slows down and plays the next two data frames. At this point, the visualization is seven data frames behind the normal playback. Finally, the visualization catches up with the normal time line.

In this widget, the distance between the black and blue dot that represents the time offset is hard to assess quickly. Moreover, this widget, as it is mainly intended to communicate the design of the time-warping function, appears next to the visualization – far away from the region of interest. Therefore, we have developed a warpometer widget that highlights the current status of the time-warp close to the selection in view space (Figure 6b) on demand. At the beginning of the selection process, both the black indicator representing the warped time t' and the blue indicator representing the global time t appear at the same location t_0. The blue indicator continues in a clockwise fashion until the end of the time window $t_0 + W_t$ is reached. The black indicator t' then behaves according to the time-warp function, and thus can move counter-clockwise, e.g. moving back in time, stop or move at a different speed than the blue indicator, e.g. when catching up.

4. Implementation

We implemented a proof of concept of our approach as an extension of an existing framework for the visualization of image and volume data, using C++, OpenGL and GLSL. For the 4D volumetric datasets, we tested the concept in combination with a regular raycaster (DVR) as well as a slice-based scattering volume renderer [PŠNB13], which was used for the biomedical datasets shown in the paper.

The time-warping function is implemented as a lookup table and stored as a texture. Depending on the size of the time window, we need to buffer a certain number of data frames. The exact frames that will be needed will be determined at runtime, after sampling the time-warping function texture. Typically, this function will not map to a stored time frame, as it does not necessarily return integer values. In that case, we simply take the nearest neighbour of the mapping. Even though linear interpolation between frames is possible in our implementation, nearest-neighbour sampling proved to be sufficient.

In our implementation, we buffer entire datasets, i.e. frames for the video streams and volumes for the volume streams. The volume renderer buffers volumes so that the user can adjust visualization parameters at all times, i.e. also after the datasets were put into the buffer. For instance, if we buffered only rendered images and the transfer function, clipping plane or a camera angle was changed after the image was put into the buffer, we would not be able to produce the correct representation using current parameters from the image alone. Another reason is that we want to keep the framework scalable. In the future, we plan to investigate more advanced composition schemes as well as experiment with projections of more than two timesteps and their representation based on the temporal offset from the global timeline. Furthermore, our implementation is sufficiently flexible to allow for the easy integration of other selection methods than brushing in image space, for example selecting the extruded space marked by the brush. Of course, GPU memory poses a limit on how many frames can be buffered, but for modern GPUs having several GBs of memory, we have not encountered any issues. However, with some limitations on composition and rendering, buffering rendered images is possible and can be useful for large-scale volumes. In our implementation, the textures are made available to the GPU via NVIDIA's bindless texture extension, i.e. textures are accessed via their integer identifier and being bound via the OpenGL context. This also gives us more flexibility regarding how many textures we can buffer.

Since the upload to the GPU remains unchanged, the rendering overhead is only introduced for rendering and composition of the two images, which can take up to twice of time. The time for upload and binding of textures remains the same, as they are accessed via the bindless extension. Finally, the images are blended on the GPU, which is a negligible performance overhead on current hardware.

5. Results

We opted to demonstrate the potential of our time-warping technique by providing examples of our localized time-warping method applied to real-world data in three usage scenarios [IIC*13]. The data were selected in order to cover a variety of application domains: video, cell biology and medical imaging. These use cases are also demonstrated in the video in the supplementary material. In these three usage scenarios, we describe various examples of time-warping.

5.1. Usage scenario 1: video monitoring

When dealing with streaming data, such as a surveillance video feed, the concept of observing the feed while allowing quick inspection of events of interest can be especially powerful.

For this usage scenario, we have downloaded video material from the VIRAT Video Dataset [OHP*11]. This surveillance video dataset was designed to be realistic and natural, as well as challenging for automatic methods in the video surveillance domain. Throughout the video, there are events happening at irregular intervals, at times multiple events at once, at other times not much is happening at all. Typical events include transferring packages between two people, cars arriving and leaving the scene, people arriving and leaving the scene and people interacting.

In this scenario, security personnel could be interested in slowly playing back events of interest locally, moments after they have happened, while still maintaining an overview on the rest of the feed in order to not miss any information from the stream. To achieve this, we designed a time-warping function that travels back in time several frames, and then plays at a slower pace than the rest of the video, finally catching up with the feed rapidly at the end. In the example in Figure 7(a), we have done exactly this in order to catch a package transfer event that we just missed.

As a second usage scenario concerning video data, we downloaded a video of a ballet performance [Hes] and focus on the comparison of the dancers' timing and movements as they perform a canon choreography technique. Canon choreography is a commonly used technique in ballet, where the dancers perform the same moves, delayed by a certain number of beats. As the choreography is known, the delay in the number of beats can be transformed to the temporal delay in frames, and an appropriate time-warp curve can be set to compare dancer's movements to train to perfection as shown in Figure 7(b). Here, we can compare the arm and head movements of the third and fourth dancers from left to right, and reveal that the left elbows are not aligned.

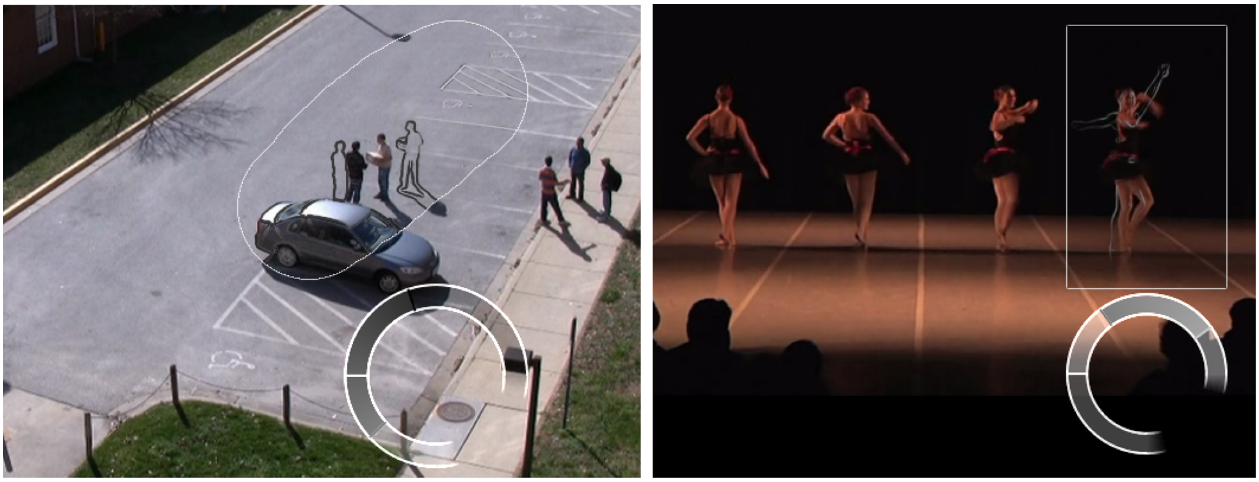
In general, the simple brushing interaction on video allows users to quickly brush over an interesting event after it has happened for a closer inspection or comparison, while the rest of the feed can still be observed. This can be useful in a surveillance context, especially when multiple camera feeds have to be observed, and the events can be sparse in the temporal domain. It is then particularly difficult for viewers to remain focused, both when many events occur simultaneously, due to limited multi-tasking capabilities, and when there are few events observed, due to fatigue. The time-warping method can help rewind and observe events, without pausing the feed. A similar scenario where this may be useful is in live sports coverage and the referee process. Here, multiple cameras observe the game, and a viewpoint decision needs to be made to find the camera that most adequately covers the event in live broadcasting. Similarly to the surveillance scenario, time-warping can be used here to rewind locally, without losing information from the current feed.

5.2. Usage scenario 2: live-cell imaging

In live-cell imaging, time-lapse microscopy is employed in order to analyse the dynamics of living cells.

For this scenario, we downloaded a 4D video of a zebrafish embryo from the Cell Image Library (CIL:36594), acquired on a Zeiss LSM 710 with a 20X 1.0 NA objective [WM]. This volumetric dataset depicts mitosis and cytokinesis, which are both parts of the cell division process over 29 time steps. The dataset features the expression of EMTB-3XGFP, which is a highly dynamic microtubule marker. High-intensity values represent high levels of fluorescence of this marker, indicating the presence of microtubules. At the centre of each cell, the mitotic spindle can be seen.

Here, researchers may be interested in focusing on what the mitotic spindle looks like exactly, but it quickly disappears after splitting. Therefore, we can employ a freezing time-warping function (Figure 8), which allows us to travel back in time and freeze the time in the selection. This way, we can examine an event after it happens,



(a) Rewinding and slowing down

(b) Delaying to synchronize a canon choreography in ballet

Figure 7: *In (a), we rewind and slow down time in order to catch an interesting package-transfer event in surveillance video footage from the VIRAT Video Dataset [OHP*11]. In (b), we delay one of the ballet dancers as they perform a canon choreography technique. The goal is to check dancer's synchronization in a video recording [Hes]. In both cases, the brushed region shows the time-warped activity as normal, as well as the current time-frame using the edge-enhancement shader, which makes people and objects within the brushed region appear in black or white outlines.*

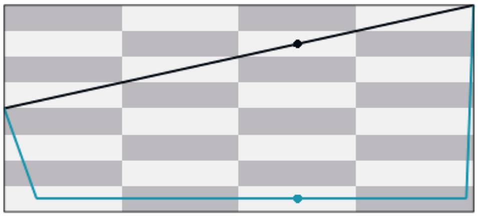


Figure 8: *In the live-cell imaging usage scenario, visible in Figure 1, we employ this the time-warping function, to travel back in time locally, in order to freeze the mitotic spindle for closer observation, while the rest of the dynamics continue as normal.*

and pause it to take a closer look, while we are still able to see the rest of the dynamics in the time series. In Figure 1(a) and (b), we can see the centre of a cell before and after mitosis. After the event happens, we can simply brush in the region of interest, and freeze the exact moment where the spindle is most clearly visible, or about to disappear (Figure 1c). If researchers want to inspect the mitosis process more slowly, we could also opt to pick a time-warping function that travels back in time, and resumes play-back from there slowly with an adjusted time-warp curve steepness.

In this scenario, we are able to pause a small region within a larger volume in order to inspect it without having to rewind and pause. In a longer time series with multiple events, this can reduce the total time it takes to inspect the whole series. Also, such an interaction allows users to quickly check if they have missed something important that requires further investigation, or if it is an event they can safely ignore. This can be done directly after an event occurs, allowing users to travel back in time taking into account their reaction time. Since the time-warping function design is flexible, we can easily customize it to individual preferences in terms of playback speed and overall behaviour.

5.3. Usage scenario 3: temporal dynamics in functional magnetic resonance imaging

In functional magnetic resonance imaging (fMRI), spatial patterns of neuronal activation as a response to a given sensory, motoric or cognitive task can be mapped based on measured changes in cerebral haemodynamics in time. These changes are often visualized as static probability maps showing which brain regions are most likely to be involved when solving the task at hand, and the resulting statistical maps are then typically shown with three-dimensional anatomical imaging in the same individual or on standardized brain templates.

Although this approach has provided unique insights into which brain regions are involved in solving various tasks, the field of research has now matured into rather investigating which networks of brain areas contribute to a given task. The networks are typically identified by their temporal characteristics, i.e. the extrinsic mode network [HRMS15], attention networks [FCS*06], salience network [MU10] and the central executive network [FZG*17] (see, for example, Lee *et al.* [LHS*12] for an overview). By far, the most studied network in absence of cognitive tasks is known as the default mode network [RMS*01, Rai10]. Up to now, the temporal characteristics of these networks, and their interplay, always assume linear evolution of time. The current usage scenario illustrates the concept of time warping in fMRI data in a simple task: bimanual finger-tapping.

Image data from one healthy volunteer were included, acquiring T1-weighted structural images, as well as a block-design BOLD fMRI at a 3Tesla GE whole body MRI system. The blocks

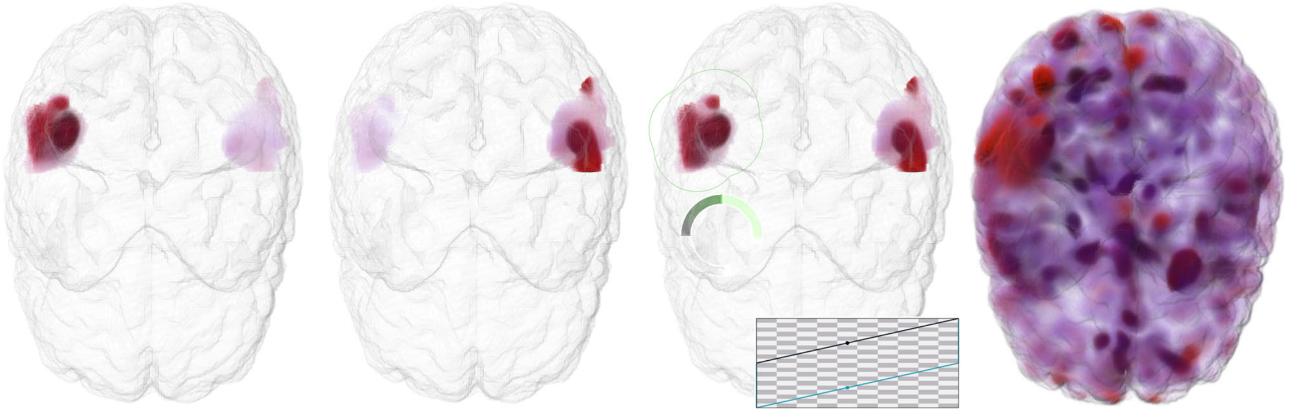

(a) Activity on the left (b) Activity on the right (c) Activity on the left delayed (d) Brain activity

Figure 9: *In the fMRI usage scenario, neuronal activity in the left motor cortex of the brain in response to right hand finger-tapping (a) is shifted to neuronal activity in the right motor cortex of the brain in response to subsequent left hand finger-tapping (b). Using our time-warping method, we are able to manipulate the dynamics of these activities so that left hemisphere activity is temporally aligned with right hemisphere activity (c) for dynamic comparison purposes. The time-warp function here instantly jumps back 10 time steps (in this case scans) to the previous scan block, and replays at normal speed. The whole brain is constantly active, and finding spatio-temporal regions of interest is difficult with automatic methods (d).*

consisted of 30 s rest (OFF), 30 s of finger-tapping with the right hand and 30 s of finger-tapping with the left hand. This sequence was repeated for a total of 100 scans, with a 10 scan block duration. After pre-processing the fMRI data (i.e. motion correction), statistical maps of neuronal activation were obtained based on general linear modelling as implemented in the SPM software package [PFA*11]. We subsequently visualized the time course of the fMRI data in the active brain regions, as well as the anatomical information from the T1 scan (Figure 9). As an illustration, the visualization is focused on the absolute voxel-wise difference between one of the resting block-scans (OFF) and the current timestep.

The goal in this usage scenario is to warp the time in the left motor cortex in order to temporally align it with activity in the right motor cortex. In the current example, by shifting the time to align the activity of two blocks using an appropriate time-warp function, comparisons between the left and right motor activity (Figure 9a and b) can be made, even if the scan paradigm did not include simultaneous finger-tapping of the left and right hand. Hence, a suitable time-warping function can be utilized to travel back in time to the start of the previous block or event, and playing back the local activity at normal speed, in order to compare the activity over time on the left and right simultaneously (Figure 9c). More generally, this could facilitate that the brain activity in one brain region can be 'frozen', while the temporal evolution in other brain regions continues to be unchanged.

Although the current example is rather simple, and therefore not directly applicable to clinical situations, it illustrates the potential of the approach to explore the interplay of various brain regions. We foresee this as a means to explore relations between networks and maybe most importantly the balance between task-dependent neuronal activity versus the network in a resting condition. This is, to our knowledge, a new way of investigating fMRI data.

While there is interest in 3D visualization of fMRI findings, for instance to show activity clusters in a 'glass brain' to communicate results [Mad15] or to visualize real-time fMRI activity [NOE*10], we are able to go one step further. Using our approach combined with more conventional 3D visualization as part of the analysis pipeline facilitates interactive exploration of the 4D data.

Although the full potential of time warping in neuroimaging research needs further exploration, the current use case shows the application to fMRI data is feasible. Visually exploring brain networks in this spatio-temporal manner may change our understanding beyond what computational methods alone can achieve.

6. Discussion

Conceptually, the specification and visualization of a time-warp volume allows for powerful operations on spatio-temporal data, and brings new possibilities, extending the basic space-time cube operations identified in the work by Bach *et al.* [BDA*17].

While powerful, designing appropriate time-functions may be challenging, and there is a learning curve associated with interpreting the results of the time-warp appropriately. This could be made more user-friendly by providing template time-warp function designs, either based on standard operations, such as rewind-freeze-speed up, or even generating appropriate time-warp functions based on events detected in the data via computer vision techniques.

The design of the time-warp function should also reflect the expected reaction time of the user in low-attention cases, where a quick reaction is needed, such as in long surveillance videos, where the observer needs to react to sparse, sudden events of short duration. An adequate time-warp function setup should go back several frames, depending on the expected reaction time of the particular user.

Currently, we have not implemented any artificial intelligence techniques that would predict users' intentions based on, i.e. the mouse movement or detected patterns in the data, and provide an appropriate time warp function automatically. Users need to specify their intention *a-priori* (slow down, delay or rewind) to get the desired effect. In some cases, such as the fMRI example, the delay might be known, so the setup of an appropriate time warp function and selecting the right time frame and spatial location is straight forward. For other cases, the users have to currently design the time warp function based on an educated guess and select regions of interest (ROI) while keeping this design in mind.

Our approach is designed for scenarios with a single static camera position and static background in mind. Camera movement and switching between multiple cameras can make the time-warp results harder to interpret, especially when a cluttered background is present in the scene. The selection may then present multiple views simultaneously, which can be confusing as the users are not necessarily able to relate the views to each other. Here also, the current edge-enhancement may not be the best choice to highlight the different time-steps. The effectiveness of our method suffers when strong camera movements are present, and even more when the captured scene contains a high level of activity with a non-uniform background. With a uniform background, as it is often the case for computer-generated scenes, small camera movements do not affect the interpretability drastically, and the edge enhancement provides enough information to distinguish the different time points. Other data abstraction techniques may be chosen to fulfil the needs of different applications.

Since rewinding and slowing down requires data buffering, another limitation of our approach is determined by the hardware – the available buffer size. As future work to circumvent this limitation, we propose to investigate importance buffering. In this case, the available buffer would be filled by data based on a priority scheme, and frames with lower importance would be skipped.

During the development of our time-warping approach, we also experimented with taking interaction time into account during the spatio-temporal region of interest selection. In this way, there could be multiple timesteps visible within a selected spatial region. Similarly, we could design the time-warp to contain multiple timesteps within a selection, which amounts to a deformation in the temporal domain with varying displacement within the spatial region. While both these options are theoretically possible, we found the interpretation of the resulting time-warp more difficult, and thus focused on warping to a single timestep at a time.

7. Conclusion and Future Work

We have presented a method for localized time-warping to select and investigate spatio-temporal regions of interest in dynamic data by designing a time-warping function and brushing a region of interest. These time-warping functions can be adjusted to fit a wide range of situations, such as localized rewinding and delayed playback, time delaying, time freezing or time slowing. The resulting selection is visualized as a focus area, while the context continues to play as normal. Aligned with the aims of visualization, tasks that our approach allows for, and that have advantages over simple rewind/slow down of video include:

- Direct interaction with data visualization parameters, e.g. live fine-tuning of parameters, such as camera angles
- Synchronization of local events in the data for comparative visualization purposes
- Interactive visualization of streaming data, including volumetric streaming data

With this technique as a foundation, we aim to improve the user interface by providing template time-warp function designs in the future, as well as more customizable visualization options, such that our technique can for instance be employed as a temporal magic lens, showing different visual representations or data aspects within the area of interest. Furthermore, we envision that time-warp functions can be derived automatically after learning from a user's own interaction behaviour and used for personalized visualization. Automatic feature detection and tracking algorithms could be used in combination with this approach so that the selection window would follow the movement features across time. As another potential research avenue, we would like to further explore the possibility of projecting spatio-temporal selections to more than one single timestep, and solving the challenges related to displaying more than two timesteps in a single image.

As it stands, our time-warping technique can be seen as a novel focus-and-context approach, which is specifically aimed at the temporal domain. This technique is generally applicable to a wide range of application domains and data types, as demonstrated in the usage scenarios, and has great potential for future applications.

Acknowledgements

The research presented in this paper was supported by the MetaVis project (#250133) funded by the Research Council of Norway. This work was partially funded by the Trond Mohn Foundation (grant number 'BFS2016TMT01'). We also thank the Bergen fMRI group and the Department of Radiology, Haukeland University Hospital for access to MRI.

References

[AAG03] Andrienko N., Andrienko G., Gatalsky P.: Exploratory spatio-temporal visualization: An analytical review. *Journal of Visual Languages & Computing 14*, 6 (2003), 503–541.

[AKK18] Atluri G., Karpatne A., Kumar V.: Spatio-temporal data mining: A survey of problems and methods. *ACM Computing Surveys 51*, 4 (2018), 83:1–83:41.

[AMM*07] Aigner W., Miksch S., Müller W., Schumann H., Tominski C.: Visualizing time-oriented data – A systematic view. *Computers & Graphics 31*, 3 (2007), 401–409.

[BBB*08] Bosco A., Bruna A., Battiato S., Bella G., Puglisi G.: Digital video stabilization through curve warping techniques. *IEEE Transactions on Consumer Electronics 54*, 2 (2008), 220–224.

[BDA*17] Bach B., Dragicevic P., Archambault D., Hurter C., Carpendale S.: A descriptive framework for temporal data visualizations based on generalized space-time cubes. *Computer Graphics Forum 36*, 6 (2017), 36–61.

[BSH*16] Bach B., Shi C., Heulot N., Madhyastha T., Grabowski T., Dragicevic P.: Time curves: Folding time to visualize patterns of temporal evolution in data. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (2016), 559–568.

[BVMG08] Balabanian J.-P., Viola I., Möller T., Gröller E.: Temporal styles for time-varying volume data. In *Proceedings of the International Symposium on 3D Data Processing, Visualization and Transmission* (Atlanta, USA, 2008), IEEE, pp. 81–89.

[FCS*06] Fox M. D., Corbetta M., Snyder A. Z., Vincent J. L., Raichle M. E.: Spontaneous neuronal activity distinguishes human dorsal and ventral attention systems. *Proceedings of the National Academy of Sciences 103*, 26 (2006), 10046–10051.

[FE17] Frey S., Ertl T.: Flow-based temporal selection for interactive volume visualization. *Computer Graphics Forum 36*, 8 (2017), 153–165.

[Fre18] Frey S.: Spatio-temporal contours from deep volume raycasting. *Computer Graphics Forum 37*, 3 (2018), 513–524.

[FZG*17] Fan J., Zhong M., Gan J., Liu W., Niu C., Liao H., Zhang H., Yi J., Chan R. C., Tan C., Zhu X.: Altered connectivity within and between the default mode, central executive, and salience networks in obsessive-compulsive disorder. *Journal of Affective Disorders 223* (2017), 106–114.

[GAW*11] Gleicher M., Albers D., Walker R., Jusufi I., Hansen C. D., Roberts J. C.: Visual comparison for information visualization. *Information Visualization 10*, 4 (2011), 289–309.

[Hes] Hess P. A.: Ballet Performance. Filmed and Edited by Peter Alfred Hess, Choreographed by Clara Ma, Music: "Are You Gonna Be My Girl" by Jet, adapted to strings by The Vitamin String Quartet. Vimeo. https://vimeo.com/18377875. Accessed 4 December 2018.

[HHWH11] Höferlin B., Höferlin M., Weiskopf D., Heidemann G.: Information-based adaptive fast-forward for visual surveillance. *Multimedia Tools and Applications 55*, 1 (2011), 127–150.

[HRMS15] Hugdahl K., Raichle M. E., Mitra A., Specht K.: On the existence of a generalized non-specific task-dependent network. *Frontiers in Human Neuroscience 9* (2015), 430.

[IIC*13] Isenberg T., Isenberg P., Chen J., Sedlmair M., Möller T.: A systematic review on the practice of evaluating visualization. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2818–2827.

[JDA07] Judd T., Durand F., Adelson E.: Apparent ridges for line drawing. *ACM Transactions on Graphics 26*, 3 (2007), 19.

[KH13] Kehrer J., Hauser H.: Visualization and visual analysis of multifaceted scientific data: A survey. *IEEE Transactions on Visualization and Computer Graphics 19*, 3 (2013), 495–513.

[LG96] Löffelmann H., Gröller E.: Ray tracing with extended cameras. *Journal of Visualization and Computer Animation 7*, 4 (1996), 211–227.

[LH11] Lampe O. D., Hauser H.: Interactive visualization of streaming data with kernel density estimation. In *Proceedings of the IEEE PacificVis* (Hong Kong, China, 2011), IEEE, pp. 171–178.

[LHS*12] Lee M. H., Hacker C. D., Snyder A. Z., Corbetta M., Zhang D., Leuthardt E. C., Shimony J. S.: Clustering of resting state networks. *PLoS One 7*, 7 (2012), e40370.

[LMWPV15] Le Muzic M., Waldner M., Parulek J., Viola I.: Illustrative timelapse: A technique for illustrative visualization of particle-based simulations. In *Proceedings of IEEE PacificVis* (Hangzhou, China, 2015), IEEE, pp. 247–254.

[Mad15] Madan C. R.: Creating 3D visualizations of MRI data: A brief guide. *F1000 Research 4* (2015), 466.

[MBG13] Mindek P., Bruckner S., Gröller M. E.: Contextual snapshots: Enriched visualization with interactive spatial annotations. In *Proceedings of the Spring Conference on Computer Graphics* (Smolenice, Slovakia, 2013), ACM, pp. 49–56.

[McC93] McCloud S.: *Understanding Comics: The Invisible Art*. William Morrow Paperbacks, 1993.

[MGB14] Mindek P., Gröller M. E., Bruckner S.: Managing spatial selections with contextual snapshots. *Computer Graphics Forum 33*, 8 (2014), 132–144.

[Mil] Mili G.: Howard Greenberg Gallery. http://www.howardgreenberg.com/artists/gjon-mili. Accessed 4 December 2018.

[MU10] Menon V., Uddin L. Q.: Saliency, switching, attention and control: A network model of insula function. *Brain Structure and Function 214*, 5–6 (2010), 655–667.

[Mül07] Müller M.: *DTW-Based Motion Comparison and Retrieval*. Springer Berlin Heidelberg, 2007, ch. 10, pp. 211–226.

[NOE*10] Nguyen T. K., Ohlsson H., Eklund A., Hernell F., Ljung P., Forsell C., Andersson M., Knutsson H., Ynnerman A.: Concurrent volume visualization of real-time fMRI. In *Proceedings of Volume Graphics* (Norrköping, Sweden, 2010), The Eurographics Association, pp. 53–60.

[OHP*11] Oh S., Hoogs A., Perera A., Cuntoor N., Chen C. C., Lee J. T., Mukherjee S., Aggarwal J. K., Lee H., Davis L., Swears E., Wang X., Ji Q., Reddy K., Shah M., Vondrick C., Pirsiavash H., Ramanan D., Yuen J., Torralba A., Song B., Fong A., Roy-Chowdhury A., Desai M.: A large-scale benchmark dataset for event recognition in surveillance video. In *Proceedings of IEEE*

CVPR (Colorado Springs, CO, USA, 2011), IEEE, pp. 3153–3160.

[PFA*11] PENNY W. D., FRISTON K. J., ASHBURNER J. T., KIEBEL S. J., NICHOLS T. E.: *Statistical Parametric Mapping: The Analysis of Functional Brain Images*. Elsevier, 2011.

[PMK*] PORTER I., MCCLELLAND S. E., KHOUDOLI G. A., HUNTER C. J., ANDERSEN J. S., MCAINSH A. D., BLOW J. J., SWEDLOW J. R.: 4D video of HeLa cells of Homo Sapiens mitosis. The Cell Image Library, CIL:13368. http://www.cellimagelibrary.org/images/13368. Accessed 4 December 2018.

[PŠNB13] PATEL D., ŠOLTÉSZOVÁ V., NORDBOTTEN J. M., BRUCKNER S.: Instant convolution shadows for volumetric detail mapping. *ACM Transactions on Graphics 32*, 5 (2013), 154:1–154:18.

[Rai10] RAICHLE M. E.: Two views of brain function. *Trends in Cognitive Sciences 14*, 4 (2010), 180–190.

[RMS*01] RAICHLE M. E., MACLEOD A. M., SNYDER A. Z., POWERS W. J., GUSNARD D. A., SHULMAN G. L.: A default mode of brain function. *Proceedings of the National Academy of Sciences 98*, 2 (2001), 676–682.

[SFP*18] SCHMIDT J., FLEISCHMANN D., PREIM B., BRANDLE N., MISTELBAUER G.: Popup-plots: Warping temporal data visualization. *IEEE Transactions on Visualization and Computer Graphics 25*, 7 (2018), 2443–2457.

[Sme14] SMESTAD G.: *Interactive Visual Analysis of Streaming Data*. Master thesis, Visualization Group, Department of Informatics, University of Bergen, Norway, 2014.

[WM] WUEHR M., MITCHISON T.: 4D video of zebrafish embryo mitosis and cytokinesis. The Cell Image Library, CIL:36594. http://www.cellimagelibrary.org/images/36594. Accessed 4 December 2018.

[WS03] WOODRING J., SHEN H.-W.: Chronovolumes: A direct rendering technique for visualizing time-varying data. In *Proceedings of Volume Graphics* (Tokyo, Japan, 2003), ACM, pp. 27–34.

[WYM08] WANG C., YU H., MA K.-L.: Importance-driven time-varying data visualization. *IEEE Transactions on Visualization and Computer Graphics 14*, 6 (2008), 1547–1554.

[ZC18] ZHOU B., CHIANG Y.-J.: Key time steps selection for large-scale time-varying volume datasets using an information-theoretic storyboard. *Computer Graphics Forum 37*, 3 (2018), 37–49.

Supporting Information

Additional supporting information may be found online in the Supporting Information section at the end of the article.

Video S1