Vol²velle: Printable Interactive Volume Visualization

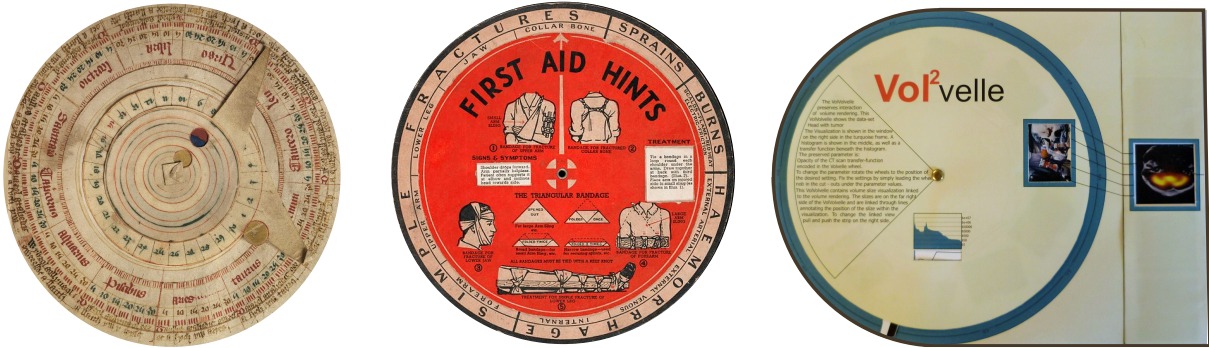Sergej Stoppel and Stefan Bruckner

Fig. 1: Volvelles, carefully designed interactive wheel charts, have been used for centuries to present a wide variety of data. Our approach allows the simple creation of a volumetric Volvelle, or Vol²velle (see rightmost image), directly from an interactive volume visualization setup. The leftmost and center image are courtesy of the National Library of Whales and the Welcome Library London (record ID = b1655695), respectively.

Abstract—Interaction is an indispensable aspect of data visualization. The presentation of volumetric data, in particular, often significantly benefits from interactive manipulation of parameters such as transfer functions, rendering styles, or clipping planes. However, when we want to create hardcopies of such visualizations, this essential aspect is lost. In this paper, we present a novel approach for creating hardcopies of volume visualizations which preserves a certain degree of interactivity. We present a method for automatically generating Volvelles, printable tangible wheel charts that can be manipulated to explore different parameter settings. Our interactive system allows the flexible mapping of arbitrary visualization parameters and supports advanced features such as linked views. The resulting designs can be easily reproduced using a standard printer and assembled within a few minutes.

Index Terms—Physical Visualization, Interaction, Volume Visualization, Illustrative Visualization

✦

1 INTRODUCTION

Volume visualization techniques for the exploration, analysis, and presentation of 3D scalar fields have been extensively studied and play an important role in many different applications such as medicine, biology, or engineering. More recently, volume visualization has also been employed in scenarios geared towards non-experts, for instance in the form of virtual exhibits in museums [19]. Interaction is often a critical component in enabling a better understanding of volumetric data through camera rotation, manipulation of the transfer function, placement of clipping planes, or the modification of light sources. When there is a need to create hardcopies of such visualizations, for instance if they are to be distributed to many people, however, the interaction aspect is lost. In this paper we present an approach for the creation of hardcopies which, to a certain extent, preserve the interactivity of the visualization. We draw our inspiration from the concept of Volvelles. A Volvelle, or wheel chart, is a type of circular slide chart. Volvelles are paper constructions with rotating parts and can be considered as an early form of an analog computer. Over the centuries, such charts have been carefully designed to accommodate scientific visualization and calculation in many different fields such as the study of planetary orbits or the presentation of multiplication tables [10]. Volvelles can be traced back to Arabic treatises on humoral medicine dating back to as early

as 1000 BC and to the Persian astronomer Abu Rayhan Biruni, who made important contributions [31]. In the 20th century, mass-produced Volvelles experienced a resurgence as handy, cheap, and easy-to-use devices for supporting organization and calculation. Today, companies such as Datalizer [1] still design and produce Volvelles as custom data visualization tools for private customers and companies like AT&T, Comcast, or Verizon.

However, the creation of a Volvelle is a laborious, time-consuming, and mostly manual process performed by graphic designers. Depending on the complexity of the encoded information, the estimated time for the design process of a Volvelle ranges from several weeks to months. In this paper, we propose an approach for the instant generation of Volvelles directly from interactive volume visualization setups to create a printable, tangible volume visualization—a VolVolvelle or *Vol²velle*—which still preserves a user-specified degree of interactivity. Clearly, paper-based visualizations are not meant to compete with handheld devices such as tablets or smartphones as a mobile solution to advanced interactive visualization. However, paper prints still have unique properties as they are very cheap, recyclable, and easily accessible, which makes them an attractive alternative for certain applications. Direct interaction with the material also increases the likelihood of remembering the encoded content [10, 38] and helps to understand the data better [37]. For instance, our approach can be used as a fast and easy way to generate personalized visualizations in museums or other public displays. In such a scenario, the user could interact with a virtual exhibit to design his or her own personalized visualization which can then be printed and assembled, creating a more engaging and memorable experience through the IKEA effect [28]. Other possible use cases include early education, product presentations, or marketing activities, where a paper-based Volvelle may also be used as an eye-catching way to distribute the URL to a fully interactive web-based visualization.

- *Sergej Stoppel is a PhD student at the University of Bergen, E-mail: sergejsto@gmail.com.*
- *Stefan Bruckner is a professor at the University of Bergen, E-mail: stefan.bruckner@uib.no.*

To the best of our knowledge, our approach is the first to employ the concept of Volvelles for the automated generation of interactive paper-based visualizations of scientific data. The main contributions of our work can be summarized as follows:

- We analyze the Volvelle design space and material constraints to identify a suitable subset for the mapping of visualization parameters.

- We introduce a set of strategies for parametrizing and encoding typical volume visualization parameters on a Vol^{2}velle.

- We present a flexible interactive approach for the semi-automatic generation of Vol^{2}velles which supports a large space of common volume visualization techniques.

Furthermore, while this paper focuses on volume data, our approach is general and can be easily extended to handle other types of visualization.

2 RELATED WORK

Research inspired by art and design has a long tradition in visualization and computer graphics. The area of illustrative visualization, for example, aims to devise computer-based visualization methods inspired by traditional illustration techniques. An overview of the concept of illustrative visualization was provided by Rautek et al. [29]. A frequent common element of these approaches is the need to extrapolate from static, paper-based presentations of phenomena to interactive visualization approaches, in particular also in the context of volume data. Examples include cutaways or ghosted views [2, 43], exploded views [4], or deformation [7]. In some sense, the work presented in this paper can be seen as completing the circle by going back to paper while preserving some of the dynamic and interactive characteristics of computer-based visualization systems.

A further source of inspiration for our work comes from research on simplifying the interaction and the specification of visualization parameters. The seminal work of Marks et al. [26] introduced Design Galleries, a general concept for exploring parameter spaces by using random sampling. They applied their approach to problems such as transfer function design in direct volume rendering and lighting specification. König et al. [20] presented a user interface paradigm for semi-automatic transfer function manipulation which provides the user with suggestions and previews. The work of Rezk-Salama et al. [30] discussed high-level user interfaces for transfer function design which integrate semantic models. Ma [25] introduced a visualization system which presents information on how parameter changes affect the result image as an image graph based on data generated during an interactive exploration process. Subsequent work by Jankun-Kelly et al. [16] deals with the exploration of visualizations, but focuses on sharing and collaboration. The work of Scheidegger et al. [32] on the VisTrails system enables the creation and retrieval of visualizations by analogy using pipeline matching based on captured provenance information. Explorable images, introduced by Tikhonova et al. [42], aim to enhance the fidelity of remote visualization by recombining a set of server-generated images to allow parameter exploration on thin clients. Our approach employs several concepts as well as technical aspects of these works. In particular, our system allows for the flexible parametrization and sampling of different visualization algorithms to create a Vol^{2}velle which can then be used to further explore the data.

Several non-traditional interfaces for visualization have been presented which are related to our research. Holman et al. [12] used projections on physical paper to create interactive information displays. Spindler discussed two approaches for tangible paper displays, Paper-Lens [33], a spatially-aware paper display that resembles a magic lens, and Tangible Views [34], paper displays for information visualization with a set of interaction techniques. Jackson et al. [15] presented a tangible interface for thin fiber structure visualization, where the movements of a paper tube were translated into interaction input. Issartel et al. [14] used tangible volumes as hand-held fish-tank displays with pressure detection in an augmented reality visualization setup. Le Goc
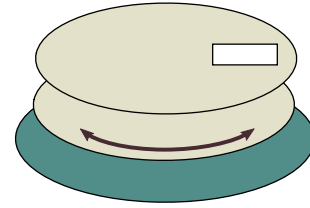

Fig. 2: The most simple Volvelle consists of a fixed front cover, a rotating wheel chart, and a fixed base.

et al. discussed design considerations for physical visualizations [22] and introduced smart tokens [21], small tangible tokens that can sense multiple types of motion. While our work also aims to create a tangible interface for interaction, we see a clear distinction. We want to preserve interaction without electronic devices and only using standard printing equipment.

A detailed chronological list of physical visualizations without electronic devices is collected in physlist [8] of which a interactive wooden model of a 3D MRI scan [9] is conceptually closely related to our approach. While physlist does not contain any interactive paper models, the works collected there share our aim of converting digital data into physical visualizations. Several works have investigated the utility of physical visualizations. Stusak et al. [39] explored the impact of physical visualization as a reward for running activities, and also discussed physical bar charts and the information retrieval process of physical visualization [35]. A similar direction was taken by Jansen et al. [18] and Taher et al. [41]. Jansen et al. [17] also presented an interaction model for visualization beyond the desktop. Hogan et al. [11] showed the strengths of a paper-based participant-aided network diagram (sociogram) for the use in a field study and even used Volvelles and slide charts for graph representations. Huron et al. [13] performed a study on how people transform data into visualizations using tangible tokens as a basis for information visualization. The supportive character of physical visualizations and their impact on memorability was evaluated by Stusak et. al. [36, 37, 38] with the result that physical visualizations can support the analysis process and lead to significantly less information decay. We see these approaches as an inspiration to use physical interaction as an alternative medium.

An example for the generation of paper-based physical visualizations is the work of Li et al. [23] who introduced an algorithm for the automatic creation of popup illustrations from 3D models. Swaminathan introduced MakerVis [40], the first tool that integrates the entire workflow for the creation of physical visualization. Yeh et al. [45] introduced Gigapixel Prints, a set of techniques for supporting large paper media with interactive input, combining the complementary affordances of paper and digital media. Cherubini et al. [6] discussed the creation, deployment, and evaluation of a large-scale, spatially-stable, paper-based visualization of a software system. While these approaches used paper as a visualization medium, they do not focus on interaction, which is a key aspect of our work.

While the work presented in this paper shares some characteristics and goals with previous approaches which mostly focus on adding tangible interaction to computer-based systems, we see a clear distinction. Our aim is to preserve interactive characteristics without electronic devices and only using standard printing equipment, as a complement rather than a competitor to high-tech solutions for tangible and mobile visualization.

3 VOLVELLE ANATOMY

Before presenting our approach for the generation of Vol^{2}velles, printed interactive volume visualizations, we want to briefly outline the possibilities and limitations of printed media. In the remainder of the paper, we use the term Volvelle to refer to wheel charts in general, while our use of the concept for the presentation of volumetric data is referred to as Vol^{2}velle. In this section we discuss the basic aspects of interactive paper models—the anatomy of a Volvelle.
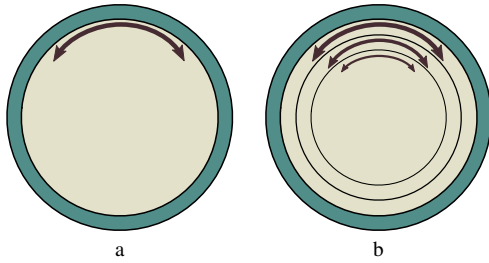
Fig. 3: The Volvelle can have either one rotating wheel (a) or multiple stacked wheels (b).
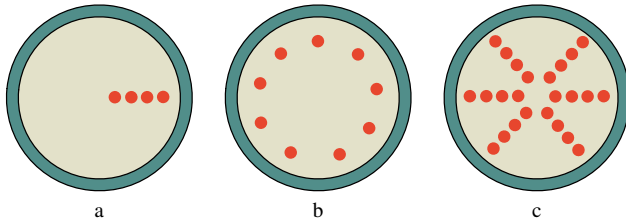

Fig. 4: The information layout on each wheel can be either concentric (a) or radial (b). Both layouts can be combined to a common concentric radial layout (c).
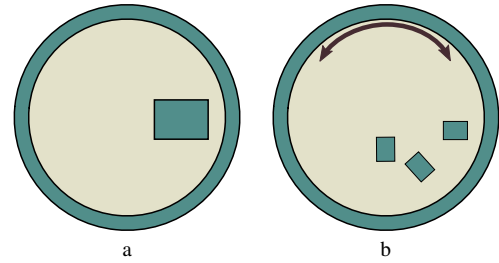

Fig. 5: A radial layout can be accomplished with a static window (a), a concentric layout requires a dynamic window (b).

Fig. 6: Going beyond the circular layout, we can employ slide charts. A slide chart consists of an envelope-like cover and a rectangular paper piece which can be pushed through the envelope.

A very basic Volvelle consists of three layers: a fixed front cover with a window, a rotating wheel, and a fixed base, as shown in Figure 2. The amount of rotating wheels may vary, but most existing Volvelles consist of only one rotating wheel as depicted in Figure 3(a) and in the examples in Figure 1. In theory, stacking the wheels allows for more interaction degrees of freedom, but practical considerations constrain this amount as more wheels require additional space (see Figure 3(b)). Moreover, each additional rotating wheel adds complexity, both in terms of assembly and with respect to the practical interaction with the final product. While the additional space requirements can be partially overcome if a transparent medium is used, the increased complexity remains a challenge. The information on one wheel can be encoded in two ways, concentrically (see Figure 4(a)) or radially (see Figure 4(b)). These layouts are not mutually exclusive and can also be combined as shown in Figure 4(c).

Concentric layouts may have dynamic windows. Hence there are two possibilities for the a cover wheel, either a static cover with a fixed window for a radial layout (see Figure 5(a)) or a freely rotating wheel with multiple windows for a concentric layout, as shown in Figure 5(b). In the latter case, the rotating wheel can be covered by a fixed front wheel with a big window, which would hide the mechanics behind it. A circular Volvelle can also be combined with other types of interactive charts, such as a linear slide chart. A slide chart consists of an envelope-like cover with windows and a rectangular piece of paper which can be pushed and pulled through, as depicted in Figure 6. Finally, it is also possible to use more complex Volvelle wheels. One example is a changing picture wheel chart as shown in Figure 7. Such a chart is constructed out of two separate wheels, which are interleaved and composed in an iris-like manner such that rotating the two wheels can be used to control which one is shown. While there are several additional possibilities, we consider the outlined components to be most useful for our purposes.

4 VOL^{2}VELLE GENERATION

Our basic approach aims at preserving the simple and straight-forward process of regular printing from a user's point of view while enabling a flexible mapping between interaction parameters and Vol^{2}velle components. We integrated Vol^{2}velle generation as an extension to an existing volume visualization framework. In this system, the complete set of

state parameters is represented by a set of properties which can be modified interactively, e.g., through mouse and keyboard commands such as camera manipulation, using user interface widgets, or programmatically. Any such modification may trigger the generation of a new output image, which can be regarded as a single sample of this parameter space. Thus, the generation of a Vol^{2}velle involves sampling a subset of this parameter space according to a user-specified set of parameters, which are then mapped to a selected Vol^{2}velle model. A conceptual overview of this process is depicted in Figure 8.

The user specifies a desired Vol^{2}velle model (see Section 4.1) which provides a number of slots for visualization parameters that can be mapped. Next, the user chooses the desired set of parameters by assigning them to the available slots. As these parameters may be of any type, our approach offers a set of automatic parametrization functions to reduce complex parameters (e.g., transfer functions or rotation matrices) to scalar values as detailed in Section 4.2. This determines the degree of interactivity that will be preserved in the Vol^{2}velle. Based on the Vol^{2}velle model and the parametrization, our system then creates a set of samples, i.e., output images of the visualization setup (see Section 4.3). After the sampling process is complete, the samples are passed to the layout generator which is responsible for arranging the images on the available Vol^{2}velle elements (see Section 4.4). This step produces the final printable document which consists of a template with cutting and assembly annotations.

4.1 Vol^{2}velle Models

While we already discussed the basic components of a Volvelle in Section 3, in this section we take a closer look at how this concept can be employed as a representation of a discretized function of multiple scalar
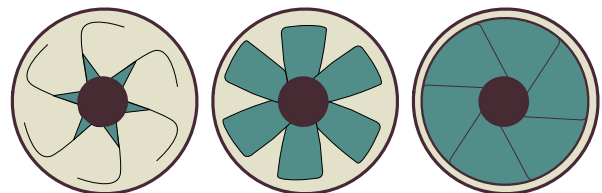

Fig. 7: By interleaving two wheel charts we can create a changing picture Volvelle which changes the appearance through rotation.
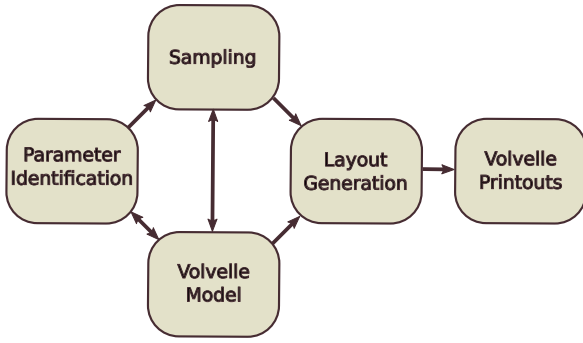
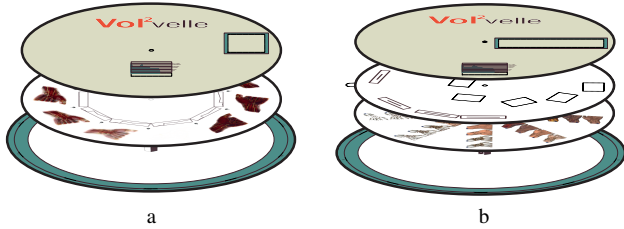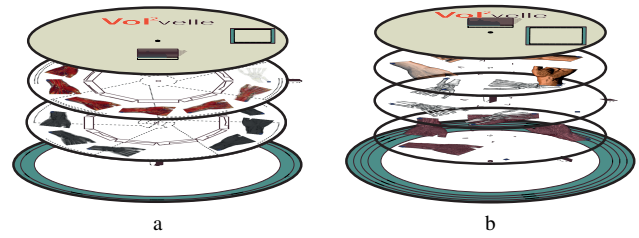Fig. 8: Conceptual overview of our approach for Vol²velle generation.

Fig. 10: (a) The wheels can be interleaved to a changing picture Volvelle. This introduces a new binary parameter but makes the assembly process slightly more complex. (b) Using a transparent medium allows superposition of multiple images.


Fig. 9: The information on the wheel can either be encoded in one (a) or two dimensions (b).

Fig. 11: The final image is composed out of individual layers.

parameters. Our approach allows users to choose between multiple Vol²velle models which constrain the number and types of interactive parameters that can be represented.

We distinguish between a *paper model* and a *transparent model*. The paper model consists entirely of paper, while the transparent model uses transparent film for the individual wheels. In the case of the paper model, we can encode one- or two-dimensional parameter spaces, through a radial (Figure 9(a)) or a radial/concentric (Figure 9(b)) layout. The radial and concentric arrangements differ in two ways. First, the number of possible radial samples is typically larger than the number of concentric samples as more samples can be arranged around a circle than along the radial line. Furthermore, the number of radial samples increases with the radius. This means that if a uniform sampling of the parameter space is desired, space on the wheel will be wasted. In addition, we offer the possibility of using a *changing picture option*, which enables the encoding of a further binary parameter. The assembly of a changing picture Volvelle is illustrated in Figure 10(a).

The transparent model consists of transparent film. This is particularly useful in the context of volume visualization, as such a *transparent model* can be used to overlay multiple independent layers. The final volume visualization can be composed similar to the multiplane camera used in early animation movies as shown in Figure 11. The resulting assembly is shown in Figure 10(b). As the order of the wheels is fixed, the presented data needs to follow the spatial ordering in the volume if correct occlusions are desired. While it would be possible to employ a decomposition similar to the work of Tikhonova et al. [42], we use a simpler approach where the layers are sorted according to their shortest distance to the eye point.

Finally, we allow a *slide chart extension* for both the paper model and the transparent model. The slide chart encodes one additional independent scalar parameter. It is used in addition to the Volvelle wheels to provide further information such as linked views. The slide chart is attached to the Volvelle base as depicted in Figure 12. A natural choice for the slide chart is an additional slice view, but our system allows the use of arbitrary 2D and 3D visualizations.

In principle, our Vol²velle allows for the representation of all three tabular data types: categorical, ordinal and quantitative data. Ordinal and quantitative data can be treated similarly, hence we simply refer to them as ordered. We can summarize the Vol²velle models in the
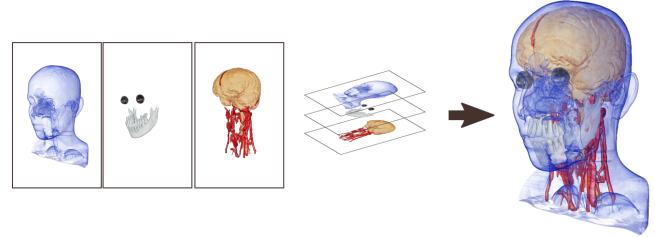
following way:

- Paper model

  – 1 Ordered Cyclic Parameter (radial layout)
  – 1 Ordered Sequential Parameter (concentric layout)
  – 1 Categorical Parameter (changing picture wheel)

- Transparent model

  – 3 Ordered Cyclic Parameters (radial layout)
  – 3 Ordered Sequential Parameters (concentric layout)

- Slide chart extension

  – 1 Ordered Sequential Parameter (slide chart)

In our system, the user can choose between the *paper model* and the *transparent model*, select the number of desired wheels and enable or disable the *changing picture* and *slide chart* options. The result is a number of slots which can then be mapped to the desired interaction parameters. This choice depends on the intended result. If the user wants an approximation of volume rendering through layer composition then a transparent model is selected. The paper model can be used more freely. However, the user has to find a suitable mapping of the parameters. For example, a mapping of the cyclic parameter to the light position and the sequential parameter to a transfer function setting is more natural than mapping the light position to a sequential parameter. We discuss the parameter mapping in detail in the next section.

4.2 Parametrization

One crucial part of our approach is the conversion of an interactive visualization setup to a discrete Vol²velle. In principle, we can regard this as a mapping between a visualization space V and a Vol²velle space V'. We characterize the spaces in Table 1. Evidently, the visualization space is by far more complex and richer than the Vol²velle space, and hence some fidelity will inevitably be lost. More formally, we can regard the two spaces V and V' as functions to an image domain and
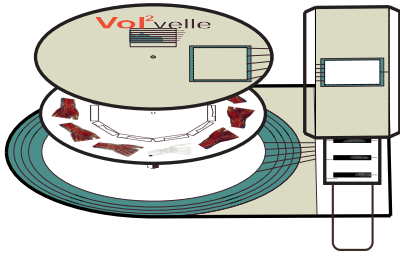
Fig. 12: A slide chart used for a linked view of volume slices.

Visualization space V	Vol^{2}velle space V'
High dimensional interaction space	Low dimensional interaction space
Continuous parameter variation possible	Discrete parametrization only
Visualization resolution/size is independent of interaction	Trade-off between the visualization size and parameter density

Table 1: A brief characterization of the Visualization space and the Vol^{2}velle space.

the Vol^{2}velle generation can be characterized as follows:

$$
\begin{array}{ccc}
V & \longrightarrow & Im \\
\uparrow F & & \downarrow S \\
V' & \longrightarrow & Im'
\end{array}
\qquad (1)
$$

As constructing a direct mapping from V' to the Vol^{2}velle image space Im' might be very complex, we subdivide the mapping into simpler parts. We first map the low dimensional Vol^{2}velle space to the complex visualization space. In the visualization space the mapping to the visualization images Im is already established. Finally, a sampling process S automatically converts the visualization image space Im according to the mapping F to the Vol^{2}velle image space Im'.

In principle, F could be constructed in a number of ways. For instance, one approach could be to employ a form of dimensionality reduction such as multidimensional scaling. However, as the user will typically have some specific communicative intent in the generation of a Vol^{2}velle, we propose a semi-automatic process which provides more control and is comparably easy to understand. We define F as:

$$
\begin{aligned}
F = (f_{Wh}^{P}, f_{Sl}) & \qquad \text{for the paper model} & (2) \\
F = (f_{Wh}^{T}, f_{Sl}) & \qquad \text{for the transparent model}
\end{aligned}
$$

where F_{Wh} denotes a mapping to the Vol^{2}velle wheel and F_{Sl} is a mapping to the slide chart. We furthermore distinguish between the paper model P and the transparent model T. In practice this means that the user has to define up to two functions f_{Wh} and f_{Sl}. The functions f_{Wh} and f_{Sl} are implemented as collections of 1D functions, i.e., the user selects parameters in V to which f_{Wh} and f_{Sl} map. Furthermore we allow some complex parameter ensembles, like settings of a transfer function, to be interpreted as a 1D parameter.

We have chosen to parametrize the Vol^{2}velle space on the unit interval. Each of these 1D function can have the parameters described in Section 4.1 as its input parameters. More formally we define:

$$
\begin{aligned}
f_{Wh,k}^{P} : \{[0,1],[0,1],\{0,1\}\} & \longrightarrow & \mathbb{R} & \quad (3) \\
f_{Wh,k}^{T} : \{[0,1]^3,[0,1]^3\} & \longrightarrow & \mathbb{R} & \quad (4) \\
f_{Sl,k} : [0,1] & \longrightarrow & \mathbb{R} & \quad (5)
\end{aligned}
$$

where k denotes the index of the 1D function. The possible input parameters of these functions are described in Section 4.1—they can be either

one ordered cyclic parameter, one ordered sequential parameter, or one categorical parameter for the paper model. The six parameters of $f_{Wh,k}^{T}$ are the three ordered cyclic and three ordered sequential parameters of the transparent model. The single parameter of $f_{Sl,k}$ is the ordered sequential parameter of the slide chart.

Generally speaking every parameter of V can be discretized with a parameter of V' and the user is free to define these mappings. However, the nature of the Vol^{2}velle parameters leads to some natural mappings. For example, a Boolean parameter in V presents a perfect match with the binary parameter of the paper model. The cyclic parameter in V' can be naturally mapped to the angle of a light position. Many natural mappings can be found between V' and V, but of course this is dependent on the specific setup of the interactive visualization. We identified a set of functions as a meaningful selection of predefined mappings applicable to many common types of parameters in volume visualization:

$$ p_i' \cdot (v_{max} - v_{min}) + v_{min} \qquad (6) $$

$$ \begin{cases} v_0: \text{for } 0 \\ v_1: \text{for } 1 \end{cases} \qquad (7) $$

$$ v_{span} \cdot sin(p_i' \cdot 2\pi) + v_{min} \qquad (8) $$

$$ v_{span} \cdot cos(p_i' \cdot 2\pi) + v_{min} \qquad (9) $$

$$ (p_i')^{v_{pot}} \cdot v_{span} + v_{min} \qquad (10) $$

where the numbers v denote user-defined values and p_i' is the i−th input variable of the function. Equation 6 is a simple linear mapping between two values v_{min} and v_{max}—such a function is well suited for a mapping to opacity values. We use this function as a preset mapping. Equation 7 defines a Boolean switch function for the changing picture Vol^{2}velle, but note that the values v_0 and v_1 are not required to be Boolean themselves. This can, for example, be used to map different light or cropping plane positions. Equations 8 and 9 can be used for cyclic mappings and are therefore well suited for the radial parameter on the Vol^{2}velle wheel. Equation 10 allows for a nonlinear mapping between two values v_{min} and $v_{max} = v_{min} + v_{span}$. Such a mapping can be used for parameters with nonlinear behavior. It is possible to create more complex functions by combining the above operations or to use multiple input variables in one mapping. An example could be a light setting on two axes which can be controlled via two parameters on the Vol^{2}velle.

In addition to these basic arithmetic functions we provide two convenience functions which treat a predefined transfer function as a 1D parameter. These functions are:

$$ Tinterpol_1(p_1', trf_1, ..., trf_k) \qquad (11) $$

$$ Tinterpol_2(p_1', p_2', trf_1, trf_2, trf_3) \qquad (12) $$

$Tinterpol_1$ linearly interpolates k transfer functions in a cyclic manner, i.e., it constructs a transition:

$$ trf_1 \rightsquigarrow trf_2 \rightsquigarrow trf_3 \rightsquigarrow ... \rightsquigarrow trf_k \rightsquigarrow trf_1. \qquad (13) $$

$Tinterpol_2$ interpolates between three transfer functions with two input parameters as:

$$ trf = \big(trf_1 \cdot p_1' + (1 - \cdot p_1') \cdot trf_2\big) \cdot p_2' + (1 - \cdot p_2') \cdot trf_3 \qquad (14) $$

The amount of possible predefined functions is of course countless. However, the presented set has proven sufficient for many common cases. The user also has the possibility to define custom functions via an integrated scripting language, which may also use one or several of the predefined functions. As all parameters have types and associated ranges, this information is used to infer a suitable default mapping (e.g., Equation 6 for floating-point parameters).

In practice, the user selects the Vol^{2}velle model and options to obtain a number of slots for which the corresponding mapping functions need to be specified. The user can then select one or several interaction parameters for each slot and, if desired, customize the mapping (e.g., by modifying the value ranges in Equation 6). Alternatively, a different predefined function may be selected or a custom function can be entered. Some additional details on this are also described in Section 5.
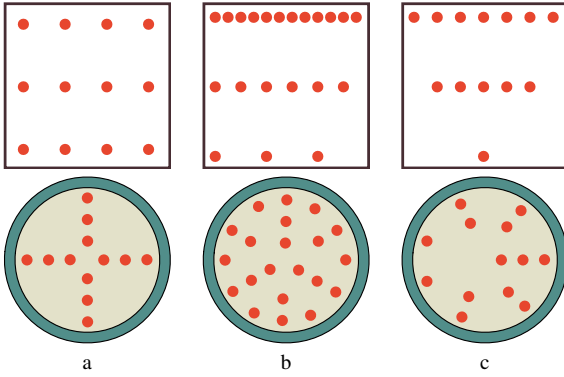
Fig. 13: The top row represents the sampling space, the bottom row represents the wheel layout. (a) A uniform distribution of the parameter, which leads to a sparse packing on the wheel. (b) Non-uniform sampling leads to a dense packing. (c) Some parameters are well suited for a conical sampling, which leads to slightly less dense packing.
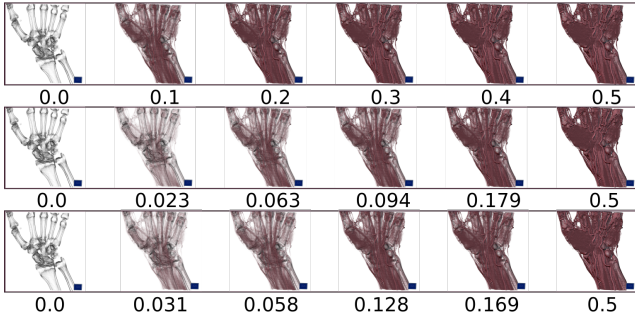

Fig. 14: **Top row:** Uniform sampling. The images look almost identical after opacity 0.2. **Middle row:** Perceptually linear parameter variation with the mean square metric. **Bottom row:** Perceptually linear parameter variation with the structural similarity metric. Both perceptually linear methods create a dense sampling for the lower transparency values. Otherwise both methods show similar results.

4.3 Sampling

During the sampling process S the function F is discretized by sampling over the parameters p'_i. The circular layout of the Vol^{2}velle introduces some constraints into the sampling process. While it would be natural to uniformly sample all parameters, this is not an ideal match for the Vol^{2}velle's geometry as it may waste large parts of the limited space on a wheel, as illustrated in Figure 13(a). In order to make use of the free space one can generate more samples with a growing distance from the center. This leads to a non-uniform sampling with a dense sample distribution on the Vol^{2}velle wheel, shown in Figure 13(b). If there is only one parameter to sample, dense sampling leads to a simple radial layout as in Figure 4(b). There are also parametrizations, for example common window level controls where a conical sampling strategy, as shown in Figure 13(c), is advantageous. As additional semantic information would be required to automatically infer an appropriate sampling pattern, the user can choose between these strategies with a default of uniform sampling. The sampling process then proceeds by generating a set of samples for each parameter p'_i of F according to the selected strategy and a chosen number of total samples. It then generates a single image for each combination of parameter values.

Particularly as a Vol^{2}velle only offers a limited capacity in terms of samples that can be placed, uniformity in parameter space may not be a good choice at all, as valuable space can be wasted with very similar images. For this reason, we also provide the option to perform perceptually uniform sampling based on the work of Lindow et al. [24]. This approach aims to equalize the distances between the output images

as measured by an image difference metric.

The linearization of one parameter is performed by first creating a dense set of uniform samples which are then used to compute pairwise image differences:

$$e(k) : K \mapsto \mathbb{R}; \qquad (15)$$
$$\text{where } e(k) = M(I(k-1), I(k)),$$

with K being the number of samples. $I(k)$ is the visualization image with the parameter setting $p(k)$, and $M()$ is an image difference metric. Our current implementation offers the standard mean square error as well as the perceptually-based structural similarity metric (SSIM) [44]. The image differences are used to create a monotonically increasing function:

$$h(k) : K \mapsto \mathbb{R}; \qquad (16)$$
$$\text{where } h(k) = h(k-1) + e(k) \text{ for } k > 0$$
$$\text{and } h(0) = e(0).$$

The function $h(k)$ represents the perceptual change of the images for a linear parameter variation. By computing the inverse function of $h(k)$ one can find a new nonlinear parametrization which leads to perceptually linear image changes according to the given metric. The linearization of multiple parameters is similarly done by first performing a dense multidimensional sampling. By fixing one parameter dimension and computing the corresponding summed differences $h_j(k, j)$ for each fixed parameter $p(j)$, and then averaging over all the $h_j(k, j)$, we compute a nonlinear parameter variation which results in a perceptually linear variation on average. For further details we refer to the work of Lindow et al. [24].

Figure 14 compares uniform sampling with perceptually-uniform sampling of a simple parameter (opacity) for the two different metrics, which in this case behave similarly.

4.4 Layout Generation

After the samples have been generated, the layout generator arranges the corresponding images on the Vol^{2}velle wheels and renders the Vol^{2}velle template as a multi-page PDF document which is then ready for printing and assembly. During layout generation, several additional graphical elements are added to each Vol^{2}velle. These include interaction guides such as tick marks and labels, elements designed to simplify the interaction, as well as additional data-dependent information. To ease the assembly, the Vol^{2}velle template uses solid lines for all graphical elements, dotted lines to indicate that cutting is required, and dashed lines where the paper should be folded.

All Vol^{2}velle models have a common basic design. Each Vol^{2}velle consists of a cover and a base which are fixed with respect to each other. The cover always contains a cutout window for the visualization, which is adjusted according to the image size. In the case of a two-dimensional sampling, the cutout window on the cover is enlarged to fit all samples. The base is mostly blank except for a ring on the edge. Labels for the parameter values are placed concentrically on the outer ring of the base wheel as shown in Figure 15(a). In order to fix the wheels at certain parameter constellations we draw cutouts directly under the value display. These cutouts are used to hold the wheels in place as shown in Figure 15(b). In the case of a changing picture
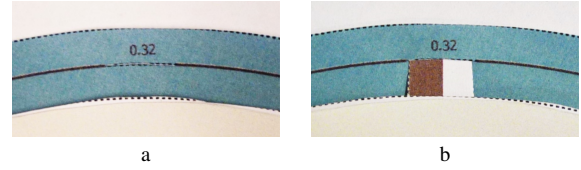

Fig. 15: (a) Values are placed radially on the base wheel. (b) In order to fix the parameter settings we add cuts under the values to fix the nobs of the rotation wheels.

Volvelle, the cutout geometry is placed over the two interchanging wheels (see Figure 21). The shape of the base is either circular or a circle with an extension to a rectangular shape for the slide chart attachment. When the slide chart option is selected, the page after the last wheel consists of the linked views. We generate the slide chart as a paper strip and a trail as shown in Figure 12. The trail has to be folded and glued onto the Vol^{2}velle base.

The common design of a Vol^{2}velle also includes a histogram of the underlying dataset on the front cover. If a transfer function is included in the parametrization, this histogram is augmented with an aligned depiction of the transfer function for the currently displayed image shown through an additional cutout. We also include the option of displaying additional text, such as a user-entered description or usage instructions, custom images, as well as meta data of the volume (if available), on the front cover.

Finally, to determine the maximum image size on the Vol^{2}velle wheels for arbitrary sampling patterns and aspect ratios, we use a force-directed layout approach.

5 IMPLEMENTATION

Our approach for Vol^{2}velle generation was implemented in C++ using the Qt toolkit as an extension to an the volume visualization framework *VolumeShop* [3] which provides several GPU-based rendering algorithms and interaction facilities. The Vol^{2}velle generation was implemented as a plugin for VolumeShop. The system includes a rich set of volume visualization methods including ranging from illustrative techniques to global illumination renderers, which are all available to the Vol^{2}velle plugin. Users can either load existing project setups, which include the specification of the desired visualization and interaction components, or create such a setup from scratch. The Vol^{2}velle module provides a simple GUI for selecting the desired model and options as discussed in Section 4.1. The user can then interactively select any parameter that describes the system's current state to assign it to a slot of the Vol^{2}velle model. The module automatically selects an appropriate parametrization function based on the type and range of the parameter, which both can be retrieved from the environment. For more complex parametrizations, such as transfer functions, the user can simply interactively modify the current system state to define the respective ranges, similar to a keyframe-based animation approach. The parametrization function can also be modified manually and the user has the ability to define custom parametrizations using an integrated scripting language.

After the user is satisfied with the parametrization, the sampling and layout process can be initiated by pressing a button and the resulting Vol^{2}velle can be viewed and printed in an integrated PDF viewer. The whole setup can be stored and re-used for other datasets or for the generation of different Vol^{2}velles, by modifying any of the non-parametrized settings, e.g., to create a Vol^{2}velle for a different viewing angle or rendering algorithm. The system supports multiple render windows and linked views for the slide chart extensions can be specified by simply selecting the desired window. This flexible design allows our Vol^{2}velle generator to be used with any existing or future visualization module of the system.

Our layout generator uses Qt's QPainter API, a powerful 2D vector graphics engine, for drawing the final Vol^{2}velle template. While currently there is only one basic Vol^{2}velle design in terms of general appearance, our implementation was performed with extensibility in mind and our software architecture allows for the straight-forward addition of alternative template designs. Additionally, certain settings such as colors or cutout positions can be exposed to the user for customization. In the future we plan to add several further templates, including ones inspired by examples of traditional Volvelles.

6 RESULTS

In this section we will present several Vol^{2}velle examples to illustrate the capabilities of our approach. The corresponding Vol^{2}velle templates are also part of the supplementary material. The time for generating the Vol^{2}velle templates is primarily determined by the specific visualization setup such as the performance of the individual rendering algorithms,
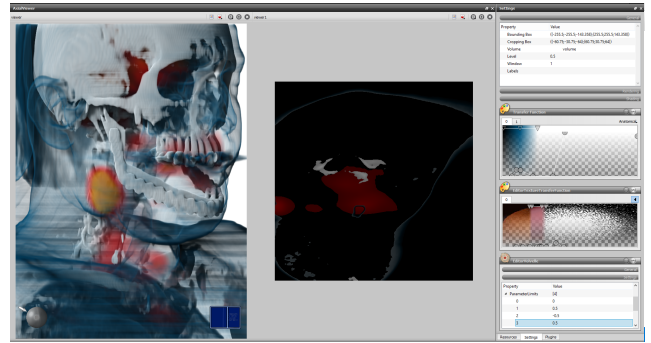

Fig. 16: A screenshot of our system for the generation of Vol^{2}velles directly from interactive volume visualization setups.
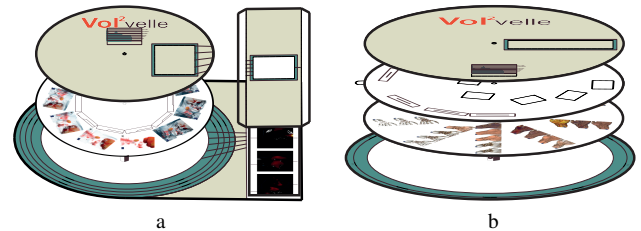

a   b

Fig. 17: (a) Linked view Vol^{2}velle with a slide chart extension. (b) Two parameter paper Vol^{2}velle with a dynamic window wheel.

but for typical GPU-based volume rendering it ranges from a few seconds (without perceptual linearization) to approximately one minute (with perceptual linearization). The time needed for the assembly of a Vol^{2}velle is of course dependent on the manual skills of the user, but an untrained person is typically able to complete the build in 10 to 15 minutes. All presented examples were printed on medium heavy paper ($120 \ g/m^2$) in A4 format. We used a simple paper fastener to hold the Vol^{2}velles together.

6.1 Linked View Vol^{2}velle

Multimodal data depicting both morphological and functional information is common in many medical applications, such as the visualization of tumors in their anatomical context. In this example, we use multimodal volume rendering of a PET-CT data set of a human head with a large tumor on the right side of the chin. A linked slice view shows detailed information about the location of the tumor.

One example for the utility of such a paper-based visualization could be patient education before an upcoming surgery. The transfer function for the PET volume is fixed, and we map the transfer function of the CT scan to the cyclic parameter of the Vol^{2}velle. By changing the opacity of the CT scan the patient could gain more insight into the nature of the shown pathological structure. As such the Vol^{2}velle could be used as part of the physician's explanations for informed consent, as a type of patient-specific pamphlet that can be produced with little effort.

The Vol^{2}velle uses a slide chart extension to present the slice view, and the axial slice position is mapped to it. The slide chart can again increase the understanding of complex data. Both parameters use a simple linear mapping with uniform sampling. A screenshot of our system which depicts the interactive setup is shown in Figure 16. The left view shows the Volume rendering and the slice view is shown on the right. The assembly of this Vol^{2}velle is illustrated in Figure 17(a). We show photographs of the finished Vol^{2}velle in Figure 18. On the left side, a photograph of the whole Vol^{2}velle is shown, and the right side depicts two different parameter settings for both opacity and slice position. On the lower right we show the volume visualization and the linked transfer function for the same parameter. It took us around 9 minutes to assemble this Vol^{2}velle.
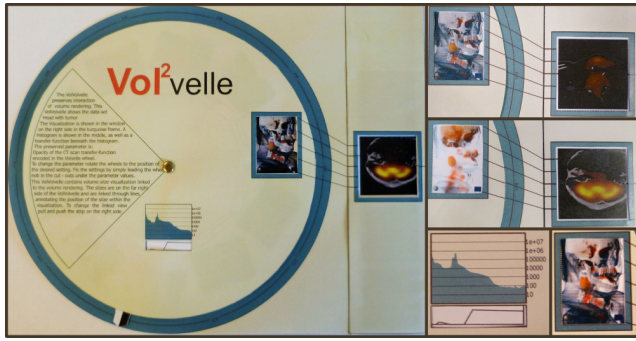
Fig. 18: Finished Vol²velle with linked views. The slice position is indicated through lines. Rotation of the wheel changes the opacity and updates the linked transfer function as shown on the lower right.
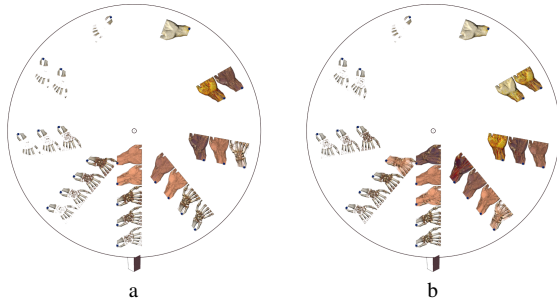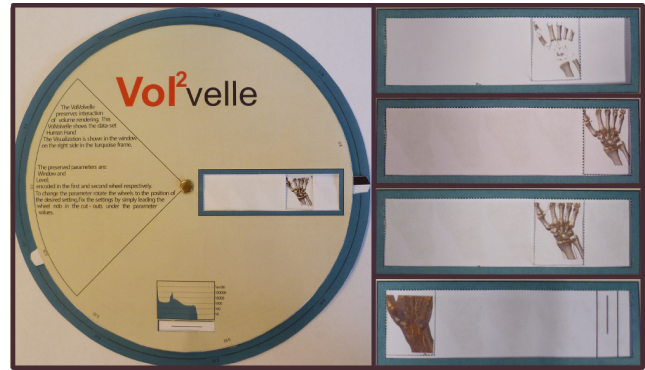

Fig. 20: A Vol²velle with two parameters on one wheel, with window as concentric and level as radial parameter. The wheel contains 25 images in total which limits the size of the images.


Fig. 19: (a): Linear sampling. (b) Perceptually linear sampling shows the change of the skin more prominently.

Fig. 21: The cutting lines intersect the images on the interleaved wheels.

6.2 Window Level Vol²velle

A common interaction scheme for the visualization of imaging data are window level controls, where the center and width of a linear ramp function are used to adjust the brightness, contrast, and/or opacity mapping of a scalar field. In this example we use a CT scan of a hand. The two parameters, window and level, respectively, are linearly mapped to the radial and angular slots of the Vol²velle using an interval of $[0.3, 0.7]$. A standard GPU-based volume renderer with an emission-absorption optical model and gradient-based shading is used.

Due to the potential non-linearity of the differences between the resulting images, this is a good example for the utility of perceptually uniform sampling. We show the resulting information wheel in Figure 19(b). In Figure 19(a) we show the same mapping without perceptual linearization, which shows rapid transitions in the visibility of the skin. The assembly of the Vol²velle is illustrated Figure 17(b). In order to be able to interact with two parameters individually, a freely rotating wheel chart with windows is placed right after the front cover.

We show a photograph of the finished Vol²velle in Figure 20, with the complete Vol²velle on the left side and detailed views of the visualization images with different settings on the right side. This Vol²velle contains 25 samples in total which leads to relatively small images, hence this layout is less suitable for setups with high-frequency changes in the images. The assembly of this Vol²velle took about 8 minutes.

6.3 Light Position Vol²velle

A very natural choice for the cyclic parameter is the position of a light source. This example uses a microCT scan of a new millipede species [27] and represents another interesting scenario for the use of Vol²velles. One could imagine the presentation of such a data set as a virtual installation in a museum or exhibition. The visitors would have several interaction options such as light position, a cutting plane to sweep through the volume, rotation of the specimen, and maybe some interaction with the transfer function. As a souvenir, the visitor has the option to create a small version of this interaction. To create
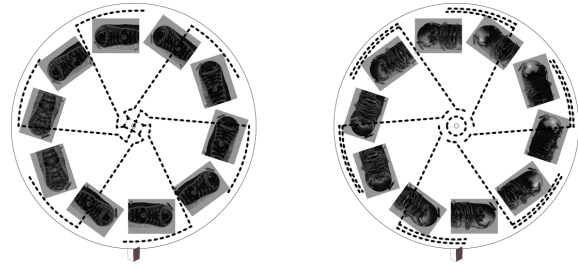
this particular Volvelle the user selects the light position as a preserved interaction and defines two positions of the cutting plane to create a changing picture Vol²velle as shown in Figure 22(a). We show photographs of the Vol²velle window in Figure 23. Some visualization images must be cut in order to employ the changing picture mechanism and the cutting lines are placed directly over the images (see Figure 21), however those cuts are only noticeable under strong light at relatively low angle to the paper.

The changing picture Vol²velle is the most complex in terms of assembly—the presented example took around 17 minutes to build. However, the additional parameter on the wheel makes this Vol²velle particularly enjoyable as the mechanics present an element of surprise.

6.4 Transparent Vol²velle

Sometimes visualization is not focused on providing an accurate representation but rather on conveying the idea of the data. This is often the case for illustrative visualization. In this example we create an expressive Vol²velle for an illustrative visualization of a human head using style transfer functions [5]. We use a segmented CT scan of a human head. The segmented parts (head, mandible, eyes, brain and blood
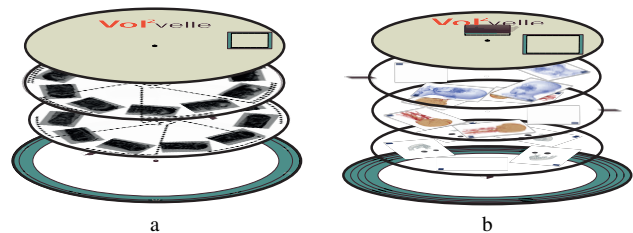

Fig. 22: (a) The assembly of a changing picture Vol²velle. The light position can be changed by rotation, and a cutting plane can be turned on or off using the iris mechanism. (b) The assembly of transparent medium Vol²velle.
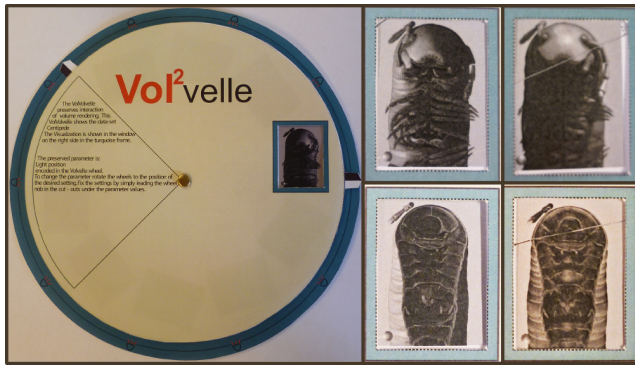
Fig. 23: The finished changing picture Vol^{2}velle. Because of the interleaved wheels, some images have to be cut.
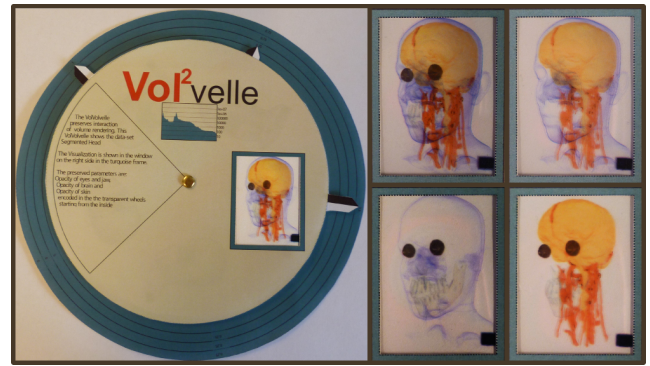

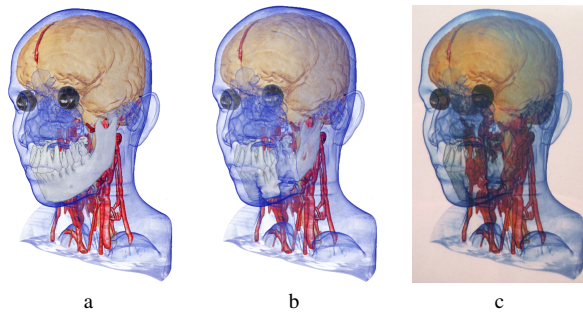Fig. 25: Finished Vol^{2}velle with transparent medium.

a   b   c

Fig. 24: (a) Direct volume rendering with style transfer functions. (b) Image-based blending of the individual layers. (c) Physical blending with transparent films. Because most printers are not capable of printing white color, the white jaw is more transparent in the physical blending.

vessels) have a fixed style with opacity as an interaction parameter. We group the mandible and the eyes as well as the brain and the blood vessels and link the Vol^{2}velle parameters to their respective opacity settings. Each of the groups is printed on a separate transparent wheel. Through the arrangement of the wheels in the order of their closest distance to the viewpoint, the composite volume rendering can be approximated. The effects of this approximation can be seen in Figure 24. The assembly of the Vol^{2}velle is illustrated in Figure 22(b). Such a Vol^{2}velle can be seen as a direct approximation of volume rendering. The user is able to change the transparency of the segmented parts resulting in an intuitive Vol^{2}velle with a pleasing appearance through the blending of individual layers.

Figure 25 shows a a photograph of the finished Vol^{2}velle. On the left we show the whole Vol^{2}velle and on the right we show detailed views for different parameter constellations. The visualization images on the Vol^{2}velle are more transparent compared to the computer generated visualization, but they still manage to convey the relationship between the individual structures well.

7   Discussion

Physical media naturally have considerable limitations in their capacity for representing complex interaction. The limitation of data storage on printed media is evident, information can only be stored on the visible parts of the printable medium. This leads to a trade off between the image size and the number of samples that can fit on a wheel. In our examples the number of images per wheel ranged from 5 to 25, Vol^{2}velles with 41 images are possible with a dense sampling. In our experience more samples are typically not practical on an A4 sheet.

In addition to space limitations, the dimensionality of the parameter space is also heavily constrained. While stacking of wheels can be used to partially overcome this, the additional complexity quickly outweighs the gain. Moreover, the full potential of stacked wheels can only be

exploited in an unconstrained design space, where the information representation is not bound to a spatial context, for instance in the case of textural information. Text snippets can be distributed over the whole Volvelle and multiple strategically placed cutouts can be used to integrate them as shown in the examples in Figure 1.

The Vol^{2}velle should be printed on white paper as most printers are not able to print white color. Many printers are not able to reproduce a rich set of gray colors, leading to rather dark images as in Figure 23. As we take a direct snapshot of the visualization for the Vol^{2}velle generation, the background of the visualization is preserved. Naturally the quality of the visualization image is dependent on the used printer, however we did not experience drastic differences between Vol^{2}velle printouts for different printers.

We allow the use of transparent media for the Vol^{2}velle generation. While transparent media potentially increase the number of possible parameters, they carry some disadvantages as well. As mentioned above most printers are not capable of printing in white color. Furthermore, prints on transparent media are fully opaque for black color only. This leads to discrepancies between the virtual and the physical visualization as depicted in Figure 24. However, there are professional printers that do not suffer from these limitations. Another issue in the context of transparent media is that our current solution requires a uniform depth ordering for the individual layers. In the future, we plan to introduce a more automatic solution for this, but the physical restriction in the number of layers means that this will still only be an approximation in most cases unless a "natural" layering exists in the underlying data.

To summarize, Vol^{2}velles can by no means be considered to be a serious competitor for tablets, augmented reality, or other advanced solutions to allow tangible interaction with visualizations and this was never our aim. However, we have shown that there are viable use cases where the low cost, simplicity, and recyclable character of a Vol^{2}velle present an interesting addition to other means of presentation. While we have focused on volumetric data and wheel charts in this paper, we also believe that other types of visualization and infographics offer interesting opportunities for further exploration of interactive paper-based interfaces.

8   Conclusion

In this paper we presented a method for the generation of Vol^{2}velles, paper-based visualizations of volumetric data with interaction capabilities inspired by traditional wheel charts. Based on an analysis of the wheel chart design space, we identified a basic set of Vol^{2}velle models which are well-suited for the presentation of volumetric data. We discussed an approach for the mapping of interactive visualizations to the limited degrees of freedom of the Vol^{2}velle and presented a flexible interactive system which enables the easy specification of this mapping. Our results show that our approach can be used to create tangible, recyclable, and interactive visualizations of volumetric data and we outlined several potential use cases. Furthermore, the concept of interactive paper-based interfaces may also stimulate future research in other areas of visualization.

REFERENCES

[1] Datalizer Slide Charts, Inc. http://www.datalizer.com/. Accessed: 2016-03-19.

[2] S. Bruckner, S. Grimm, A. Kanitsar, and M. E. Gröller. Illustrative context-preserving exploration of volume data. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1559–1569, 2006.

[3] S. Bruckner and M. E. Gröller. VolumeShop: An interactive system for direct volume illustration. In *Proc. Visualization*, pages 671–678. IEEE, 2005.

[4] S. Bruckner and M. E. Gröller. Exploded views for volume data. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1077–1084, 2006.

[5] S. Bruckner and M. E. Gröller. Style transfer functions for illustrative volume rendering. *Computer Graphics Forum*, 26(3):715–724, 2007.

[6] M. Cherubini, G. Venolia, and R. DeLine. Building an ecologically valid, large-scale diagram to help developers stay oriented in their code. In *Proc. VL/HCC*, pages 157–162. IEEE, 2007.

[7] C. D. Correa, D. Silver, and M. Chen. Constrained illustrative volume deformation. *Computers & Graphics*, 34(4):370–377, 2010.

[8] P. Dragicevic and Y. Jansen. List of physical visualizations. www.dataphys.org/list, 2012. Last accessed Feb 2015.

[9] N. Fraser. Rearrangeable wooden model of brain scan. www.dataphys.org/list, 2008. Last accessed Jan 2013.

[10] J. Helfand. *Reinventing the Wheel*. Princeton Architectural Press, 2006.

[11] B. A. Hogan and B. Juan Wellman. Visualizing personal networks: Working with participant-aided sociograms. *Field Methods*, 19(2):116–144, 2007.

[12] D. Holman, R. Vertegaal, M. Altosaar, N. F. Troje, and D. Johns. PaperWindows: Interaction techniques for digital paper. In *Proc. CHI*, pages 591–599. ACM, 2005.

[13] S. Huron, Y. Jansen, and S. Carpendale. Constructing visual representations: Investigating the use of tangible tokens. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2102–2111, 2014.

[14] P. Issartel, F. Guéniat, and M. Ammi. A portable interface for tangible exploration of volumetric data. In *Proc. VRST*, pages 209–210. ACM, 2014.

[15] B. Jackson, T. Y. Lau, D. Schroeder, K. C. Toussaint, and D. F. Keefe. A lightweight tangible 3D interface for interactive visualization of thin fiber structures. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2802–2809, 2013.

[16] T. J. Jankun-Kelly, K.-L. Ma, and M. Gertz. A model and framework for visualization exploration. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):357–369, 2007.

[17] Y. Jansen and P. Dragicevic. An interaction model for visualizations beyond the desktop. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2396–2405, 2013.

[18] Y. Jansen, P. Dragicevic, and J.-D. Fekete. Evaluating the efficiency of physical visualizations. In *Proc. CHI*, pages 2593–2602. ACM, 2013.

[19] D. Jönsson, M. Falk, and A. Ynnerman. Intuitive exploration of volumetric data using dynamic galleries. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):896–905, 2016.

[20] A. König and M. E. Gröller. Mastering transfer function specification by using volumepro technology. Technical Report TR-186-2-00-07, Vienna University of Technology, 2000.

[21] M. Le Goc, P. Dragicevic, S. Huron, J. Boy, and J.-D. Fekete. SmartTokens: Embedding motion and grip sensing in small tangible objects. In *Proc. UIST*, pages 357–362. ACM, 2015.

[22] M. Le Goc, P. Dragicevic, S. Huron, and J.-D. Fekete. Design considerations for composite physical visualizations. In *Proc. CHI*. ACM, 2015.

[23] X.-Y. Li, C.-H. Shen, S.-S. Huang, T. Ju, and S.-M. Hu. Popup: automatic paper architectures from 3D models. *Transactions on Graphics*, 29(4):111:1–9, 2010.

[24] N. Lindow, D. Baum, and H.-C. Hege. Perceptually linear parameter variations. *Computer Graphics Forum*, 31(2):535–544, 2012.

[25] K.-L. Ma. Image graphs - a novel approach to visual data exploration. *Proc. IEEE Visualization*, pages 81–513, 1999.

[26] J. Marks, B. Andalman, P. A. Beardsley, W. Freeman, S. Gibson, J. Hodgins, T. Kang, B. Mirtich, H. Pfister, W. Ruml, K. Ryall, J. Seims, and S. Shieber. Design galleries: A general approach to setting parameters for computer graphics and animation. In *Proc. SIGGRAPH*, pages 389–400. ACM, 1997.

[27] B. D. M. Nesrine Akkari, Henrik Enghoff. A new dimension in documenting new species: High-detail imaging for myriapod taxonomy and first 3D cybertype of a new millipede species (diplopoda, julida, julidae). *PLoS ONE*, 10(8):e0135243, 2015.

[28] M. I. Norton, D. Mochon, and D. Ariely. The IKEA effect: When labor leads to love. *Journal of Consumer Psychology*, 22:453–460, 2012.

[29] P. Rautek, S. Bruckner, E. Gröller, and I. Viola. Illustrative visualization: New technology or useless tautology? *SIGGRAPH Computer Graphics*, 42(3):4:1–4:8, 2008.

[30] C. Rezk-Salama, M. Keller, and P. Kohlmann. High-level user interfaces for transfer function design with semantics. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1021–1028, 2006.

[31] J. Russell and R. Cohn. *Volvelle*. Book on Demand, 2015.

[32] C. Scheidegger, H. Vo, D. Koop, J. Freire, and C. Silva. Querying and creating visualizations by analogy. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1560–1567, 2007.

[33] M. Spindler, S. Stellmach, and R. Dachselt. PaperLens: Advanced magic lens interaction above the tabletop. In *Proc. ITS*, pages 69–76. ACM, 2009.

[34] M. Spindler, C. Tominski, H. Schumann, and R. Dachselt. Tangible views for information visualization. In *Proc. ITS*, pages 157–166. ACM, 2010.

[35] S. Stusak and A. Aslan. Beyond physical bar charts: An exploration of designing physical visualizations. In *Proc. CHI*, pages 1381–1386. ACM, 2014.

[36] S. Stusak and A. Butz. Can physical visualizations support analytical tasks? In *Proc. VIS Posters*. IEEE, 2013.

[37] S. Stusak, M. Hobe, and A. Butz. If your mind can grasp it, your hands will help. In *Proc. TEI*, pages 92–99. ACM, 2016.

[38] S. Stusak, J. Schwarz, and A. Butz. Evaluating the memorability of physical visualizations. In *Proc. CHI*, pages 3247–3250. ACM, 2015.

[39] S. Stusak, A. Tabard, F. Sauka, R. A. Khot, and A. Butz. Activity sculptures: Exploring the impact of physical visualizations on running activity. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2201–2210, 2014.

[40] S. Swaminathan, C. Shi, Y. Jansen, P. Dragicevic, L. Oehlberg, and J.-D. Fekete. Creating physical visualizations with MakerVis. In *Proc. CHI Extended Abstracts*, pages 543–546. ACM, 2014.

[41] F. Taher, J. Hardy, A. Karnik, C. Weichel, Y. Jansen, K. Hornbæk, and J. Alexander. Exploring interactions with physically dynamic bar charts. In *Proc. CHI*, pages 3237–3246. ACM, 2015.

[42] A. Tikhonova, C. Correa, and K.-L. Ma. Explorable images for visualizing volume data. In *Proc. PacificVis*, pages 177–184. IEEE, 2010.

[43] I. Viola, A. Kanitsar, and M. E. Gröller. Importance-driven feature enhancement in volume visualization. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):408–418, 2005.

[44] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[45] R. B. Yeh, J. Brandt, J. Boli, and S. R. Klemmer. Interactive gigapixel prints: Large paper interfaces for visual context, mobility, and collaboration. Technical report, Stanford University, 2006.