

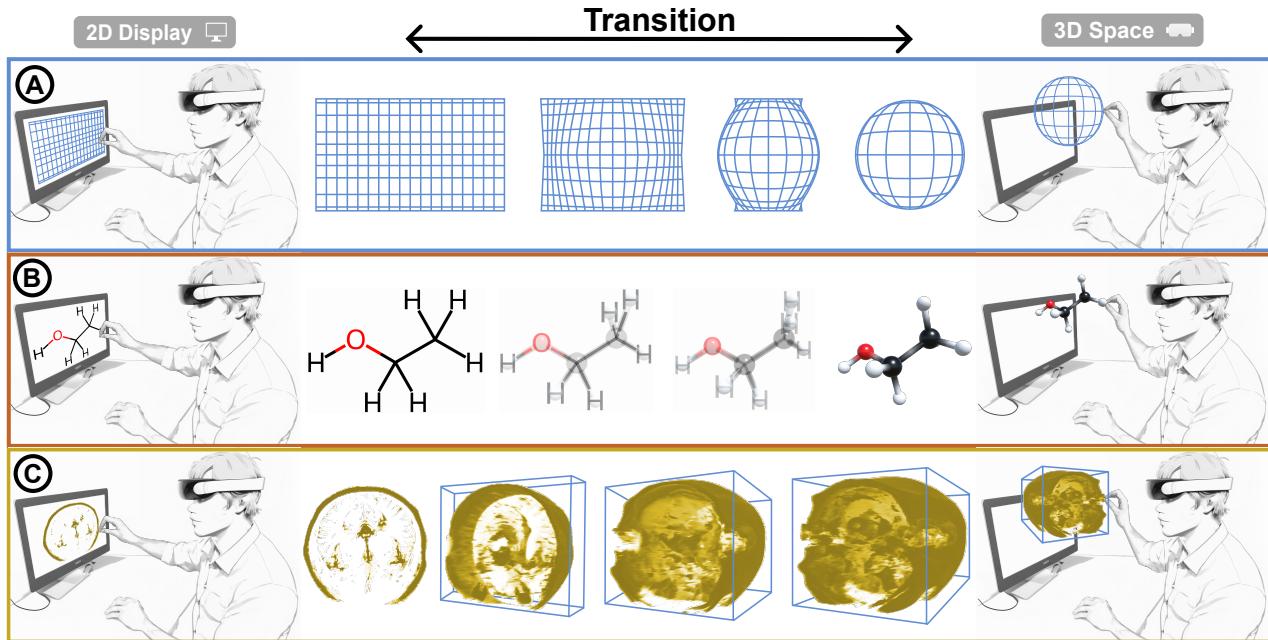
# A Design Space for Visualization Transitions of 3D Spatial Data in Hybrid AR-Desktop Environments

Yucheng Lu (陆禹承)<sup>1</sup> Tobias Rau<sup>1,2</sup> Benjamin Lee<sup>3</sup> Andreas Köhn<sup>2</sup> Michael Sedlmair<sup>2</sup> Christian Sandor<sup>1</sup> Tobias Isenberg<sup>1</sup>

<sup>1</sup> Université Paris-Saclay, CNRS, Inria, LISN, France

<sup>2</sup> University of Stuttgart, Germany

<sup>3</sup> JPMorganChase, USA



**Figure 1:** Schematic views of XR transitions between 2D and 3D representations, applied to (A) a planar projection and its intermediate states, (B) a structure formula of a molecule and its 3D ball-and-stick representation, and (C) an MRI slice and its full volumetric representation.

## Abstract

We present a design space for animated transitions of the appearance of 3D spatial datasets in a hybrid Augmented Reality (AR)-desktop context. Such hybrid interfaces combine both traditional and immersive displays to facilitate the exploration of 2D and 3D data representations in the environment in which they are best displayed. One key aspect is to introduce transitional animations that change between the different dimensionalities to illustrate the connection between the different representations and to reduce the potential cognitive load on the user. The specific transitions to be used depend on the type of data, the needs of the application domain, and other factors. We summarize these as a transition design space to simplify the decision-making process and provide inspiration for future designs. First, we discuss 3D visualizations from a spatial perspective: a spatial encoding pipeline, where 3D data sampled from the physical world goes through various transformations, being mapped to visual representations, and then being integrated into a hybrid AR-desktop environment. The transition design then focuses on interpolating between two spatial encoding pipelines to provide a smooth experience. To illustrate the use of our design space, we apply it to three case studies that focus on applications in astronomy, radiology, and chemistry; we then discuss lessons learned from these applications.

## CCS Concepts

- **Human-centered computing** → Visualization theory, concepts and paradigms;

## 1. Introduction

A myriad of domains and applications depend on 3D spatial data for their investigations. While 3D spatial data is presented in stereo-

scopic 3D views without loss of structural information, we need to resort to some form of projection when we display it on 2D screens such as computer monitors. The projections commonly used in computer graphics simulate a (pinhole camera) view of 3D spatial data,

but they struggle with providing viewers with a proper depth impression, particularly in data visualization. Other projection methods and transformations can convert the 3D spatial data into a 2D visualization. A map projection [Jen12], e.g., facilitates an occlusion-free view of the surface of the Earth or, combined with other transformations, can give an overview of the surface of biomolecules [KFS\*17]. In other application scenarios, abstract 2D representations are used to increase readability and emphasize information such as topology. Structural formulas of molecules, for instance, are closer to graphs than to projections. Suitable transformations are thus usually determined by the domain, the dataset, and the application. Such transformations, however, always bring drawbacks, primarily distortion and filtering information for the sake of readability. Being able to switch between a projection on a traditional 2D display and a stereoscopic view of the 3D dataset could combine the strengths of both (e.g., the ability to see the whole surface but also getting an undistorted sense of distances, angles, and directions).

Hybrid (mixed reality) environments [FS91, BBK\*06] allow users to interact with multiple systems that reside at different points on the reality-virtuality continuum (RVC) [MTUK95] at the same time. The combination of an immersive (e.g., AR, VR) and a non-immersive (e.g., desktop workstation, tablet) environment can display 3D and 2D visualizations in the environment that they were originally designed for: e.g., 2D views on a desktop alongside visualizations of 3D spatial data in AR. Another benefit is that these hybrid systems support or even enhance established desktop workflows that involve visual data analysis and even data manipulation on 3D spatial data via 3D input and stereoscopic rendering [CDH\*19, FAP\*22, RSK24, RKP\*24, LCKP25], without losing features of specialized desktop applications and the familiar input. The AR-Desktop hybrid system can thus be seen as providing the “best of both worlds,” while also extending the available workspace (Figure 1). Visualizations in different manifestations of the RVC that the AR and desktop setup offers us (also called actualities [AGF\*23]), when simultaneously shown in a juxtaposed fashion, can provide a better data overview and improve the data-understanding process [WBR\*20]—compared to a desktop-only setup. The actual *mental* transformation to connect a 2D with a 3D representation of the same data, however, is difficult due to the data’s complexity and the differences of the representations, and the most beneficial 2D and 3D visual representations may differ substantially.

While juxtaposing multiple 2D and 3D visualization mappings can compensate individual shortcomings, a smooth transition within a single visualization can provide a clearer connection between 2D and 3D representations. It is particularly beneficial when users wish to focus on interaction and intermediate transitional states, such as sliced volumes. One possible way to establish clearer connections between 2D and 3D representations in a single visualization is the use of animations [TMB02]. In our case, they provide means for enhancing the mental connection between structural information in 2D and 3D and, consequently, reducing cognitive load. Studies indicate that animated transitions between different actualities can provide users with a clear picture of a given projection’s distortion and can establish a connection between the two (or more) representations [SPZS24]. Animations also improve people’s understanding of the relationship between different visual mappings [HR07].

When exploring ways of mapping spatial datasets to 2D and 3D representations, placing and scaling these visualizations for an immersive and non-immersive environment, adding annotations, and animating the transition between the representations, the multitude of possibilities quickly becomes unmanageable. We thus propose a design space to better understand and manage possible considerations. While previous design spaces [LCP\*22, HHS\*24] have demonstrated the potential for representing abstract information in Mixed Reality, we focus on actuality-adaptive visualization transitions within hybrid environments, specifically for spatial datasets.

To develop the design space, we first explored possible datasets and domains that provide data with high complexity in order to benefit from the proposed approach. In addition, the datasets have to provide at least one 2D and one 3D representation that is well-known in their respective domains. We then developed a prototype that implements the transition process of these meaningful representations—focusing on 2D and 3D visualizations of spatial datasets in a hybrid AR-desktop environment. This prototype helped us to refine our design dimensions and to verify and showcase the design space with three example datasets (Exoplanet, Brain, Molecule).

In the first part of our design space we provide a description of the initial and the target visualization, which consists of three main parts: (1) the *visual mapping* that describes transformation and the resulting dimensionality of the representation, (2) *annotations* that provide supporting indicators with the goal of increase readability, and (3) *pose in hybrid environment* that defines where the visualization is located and how it is scaled. In the second part, we describe possible design dimensions when animating the interpolation between the initial and the target visualization. With these elements we can describe a variety of visualization presentation setups in the AR-desktop environments and apply seamless *transitions* between different visualization setups with interpolation and animation. In addition, we further discuss considerations that focus on adopting the design space to a wide range of 3D spatial data (Section 6). With the design space and the discussion of high-level design considerations, we aim to inspire the creation process of hybrid systems with animated actuality adaptive visualization transitions.

To illustrate the utility of our design space, we use our prototype to demonstrate this system as well as our design space with three case studies (Section 5): (i) astrophysics data with star systems that have exoplanets; (ii) magnetic resonance imaging (MRI) scan of a human brain; and (iii) molecular structures of different complexity.

In summary, we contribute a design space for 3D spatial data visualization transitions and lessons learned from three case studies with datasets from astrophysics, medical visualization, and chemistry.

## 2. Related work

Our work describes transitions of visualizations that change their dimensionality corresponding to the environment. In particular, we work with a hybrid system that uses AR and desktop PC environments and relies on projection and animation, as we discuss next.

### 2.1. Cross-reality systems

Many approaches can facilitate a seamless blend of different actualities, which have been surveyed by Auda et al. [AGF\*23]. For

example, users can *transition* from one actuality to another, which is also known as a transitional interface (TI) [BKP01]. From the three types of systems that Auda et al. derived, however, the *substitutional* one is the most relevant for us as we focus on scenarios in which a user controls virtual objects that transition between different environments [LCP\*22, LSC\*23, SPZS24, RIK\*25]. Fröhler et al. [FAP\*22], e.g., use *cross-virtuality* to describe visualization applications that transition between different actualities, and Zagermann et al. [ZHB\*22] think of cross-device interaction, like ours, as *complementary interfaces*—in our case, the AR and the desktop complement each other to take advantage of both 2D and 3D spaces.

Existing literature mainly focuses on interaction design. A design space of single users interacting with a hybrid system was proposed by Wang and Maurer [WM22]. In one of their scenarios, a user moves a visualization from one point of the RVC to another, which involves a transformation of the visualization from 2D to 3D. A similar scenario was implemented and studied by Schwajda et al. [SFP\*23] using graph data that transitions from a large-scale display into an AR environment. Another implementation by McDade et al. [MJC23] renders a 3D model as an exploded view on the PC and as a merged model in AR. While these approaches [GFM\*22] focus on the aspects for supporting users to form a joint mental model of both visualizations, Lee et al. [LCP\*22] formulated a general design space for systems that incorporate abstract data visualization transitions from 2D to 3D and vice versa in a single actuality. In our work we focus on spatial data visualizations that the design space of Lee et al. [LCP\*22] did not consider or could not support.

We also note that, in a AR-desktop hybrid environment, augmenting 2D content in desktop monitors with 3D AR content is a common approach. Gall et al. [GHFH23], e.g., enhance the uncertainty visualization of the distribution by providing extra 3D AR visualization and facilitating gestural interaction. They and others apply this approach to tomography visualization [GFM\*22, GHW\*24, MNT\*24]. Other studies focus on the interaction techniques for transferring virtual objects from desktop monitors into the AR space. Cools et al. [CGS\*22] propose a framework to expand the desktop monitor to AR space through an arced virtual screen. Rau et al. [RIK\*25] explore various forms of gestural interaction that enable users to bring desktop monitor content into AR space. All of these works also show that an AR-desktop hybrid can complement the shortcomings of traditional desktop visualization, yet the spatial visualization transition design is still unclear—which is what we add.

## 2.2. Projections and mappings

For further discussion of the related work and, later, the design space we need to define three terms. First, *transformation* is an operation, e.g., a projection or an algorithm, that maps a visualization between 2D and 3D. Second, *transition* is the process of changing the actuality of a visualization. In our specific case, we change the visualization’s environment from 2D to 3D or from 3D to 2D. Third, *morph* is the continuously animated shift between visualization representations. In general, a morph is independent of the environment in which the visualization is rendered, but for our proposed design space we consider that a transition always incorporates a morph.

Visualizations of 3D spatial data or descriptions based on the 3D physical world [CMS99] facilitate the recall of the physical spatial

understanding. Nowadays, 3D visualizations are still predominantly rendered and analyzed on traditional PC workstation setups that project the final image on 2D displays [HRD\*19]. Visualizing 3D datasets on 2D screens, however, suffers from occlusion [LGV\*16], and perspective distortion [PPZ\*12]. For a good spatial understanding of a perspective view of complex 3D visualizations, users have to navigate the scene [DE98], which can be difficult using standard workstation input modalities. Hurter et al. [HRD\*19] directly compared 3D visualizations projected on a 2D screen with the immersive space, and suggested that the immersive visualization “fosters the discovery of many additional insights.”

2D projections of 3D data, in contrast, can be more comprehensible than 3D visualizations [NVV\*06, BGP\*11]. Geometric and abstract projections are commonly used to simplify the depiction of 3D spatial data, such as Earth map projection methods in geography [Sny87, Jac05, Jen12]. While distortion is inevitable due to non-isometric transformations, projections retain specific visual characteristics. The Azimuthal Equidistant projection [And74], e.g., keeps the distance undistorted when projecting the globe onto a plane. In medicine, conformal mapping is widely used to locally preserve small visual shapes by keeping internal angles invariant [KMM\*18]. Physical 2D screens are considered precise and fast for 2D content, but limiting for 3D content [BSB\*18]. Although it is debated whether 2D outperforms 3D or vice versa [TKAM06, HMK\*19, LNP\*23], hybrid interfaces are a valuable option that benefits from both, 2D and 3D [DRST14, HHS\*24]. Users tend to make use of both 2D and 3D visualizations to complete their tasks depending on the circumstances [NVV\*06], thus combining their advantages [MVB\*17]. Certain 2D and 3D visual representations are inherently more compatible with the affordances of 2D screens and 3D AR environments, and are thus particularly suitable for hybrid settings. Building on these foundations, we develop our hybrid AR–desktop environment and the design space.

## 2.3. Animation and transition

Animation transitions have been widely adopted in various applications, often to facilitate spatial understanding during transitions between Augmented Reality (AR) and Virtual Reality (VR) [PFKJA24, LCKP25] or to help users track changes between visualizations [EDF08]. Liao et al. [LCKP25], e.g., proposed an approach to smoothly morph 3D models from a 2D to a 3D representation between AR space and the monitor. They show that the animated transitions enhance user engagement and support perceptual tracking of objects across different representations. In our work, the transitions occur between AR and desktop environments, accompanied by changes in the visualization itself. We focus on transitions that adapt to the AR space or physical monitor displays as well as on animation designs used to handle changes in visual encoding.

To design the animation, several common designs already exist, such as animation curves [DBJ\*11], staggered animations, or approaches for controlling animations [ARES13]. We include some of these established designs and further add an additional interpolation method dimension for the design of animations in transitions.

### 3. Design Considerations

Below, we begin by briefly summarizing our overall process for arriving at the design space, including our core considerations for 2D and 3D representations of spatial visualizations and the resulting design of transition animations.

#### 3.1. Development process

Spatial data presents inherent positional information that necessitates careful consideration of how to encode it into visual representations. This positional information is also essential in the design of animations, particularly when transitioning between 2D and 3D visualizations. To develop our proposed design space, we centered the transition design around the concept of spatial information.

We started by reviewing relevant literature on foundational concepts and looked for datasets and domains that are well-suited as examples and potential applications of the design space. We found that datasets in the domains of medicine, geography, and chemistry cover a wide range of scenarios that we want to support with our design space. Each dataset has an inherent 3D representation and at least one unique 2D mapping (here: projection, slice, and graph representation) that is wildly different from the other data domains and their respective 3D representation. Further, prior research has summarized design choices for 2D and 3D visualizations [HHS\*24], design space for transitioning non-spatial data [LCP\*22], and various geometry projection techniques for 2D and 3D visualizations, such as flattening or projection techniques [KMM\*18, KFS\*17, MC01, Sny87]. The chosen domains and studies encompass a wide range of data types, including volumetric data, point-based data, and topological data, which are frequently encountered in various scientific and engineering domains. Next, we examined existing systems that design and implement transitions within XR environments [HRD\*19, MDLI\*18, HMK\*20, MAB\*18, YJD\*18, SFP\*23].

While Lee et al. [LCP\*22] explored design choices for non-spatial datasets and interaction, our interest lies in how 3D data is expressed based on its inherent spatial characteristics and how it is maintained during the transition, within the hybrid AR-desktop environment. To realize this goal, we extend their existing visualization design, but deliberately omit specific design dimensions such as input modalities or other devices. We focus on the transition relationship between the physical screen and AR space and thus generalize our design space based on the exploration of the three mentioned datasets.

#### 3.2. Intermediate Visualization

Transitions in visualization involve changes in specific design dimensions. A common scenario in AR-desktop environments is to transition from a 3D model to a flattened 2D view by altering the positional encoding [KMM\*18]. Yet each individual visualization is produced through a set of design decisions, so transitions primarily reflect a shift between two distinct visualizations or two sets of design choices. We thus focus on designing these transitions to manage the changes between two sets of design choices.

Transitions between design choices, however, may also produce intermediate visual representations, which result naturally from the interpolation during transitions between the two sets of designs.

Many, if not most, of these intermediate visual representations only facilitate a visually smooth animation but are semantically insignificant: they do not necessarily represent an actual state of the data as they show a distorted intermediate state, which likely reduces the visualization's readability. In certain cases, however, the intermediate representations can be meaningful if they shift the presentation to reveal spatial data from a different perspective (as we later, in Section 5, demonstrate in our case study on volumetric data). Nonetheless, the specific sequence and animation of the intermediate designs can have a substantial impact on the effectiveness of the transition.

### 4. Design Space

Based on these considerations, we separate our transition design into two parts: (1) the design of the initial and the final visualization (Figure 2) and (2) the design of the transition including intermediate states between two sets of design choices (Figure 3). The parameters we discuss within these two parts describe possible degrees of freedom for variation of the specific transition, as we describe next.

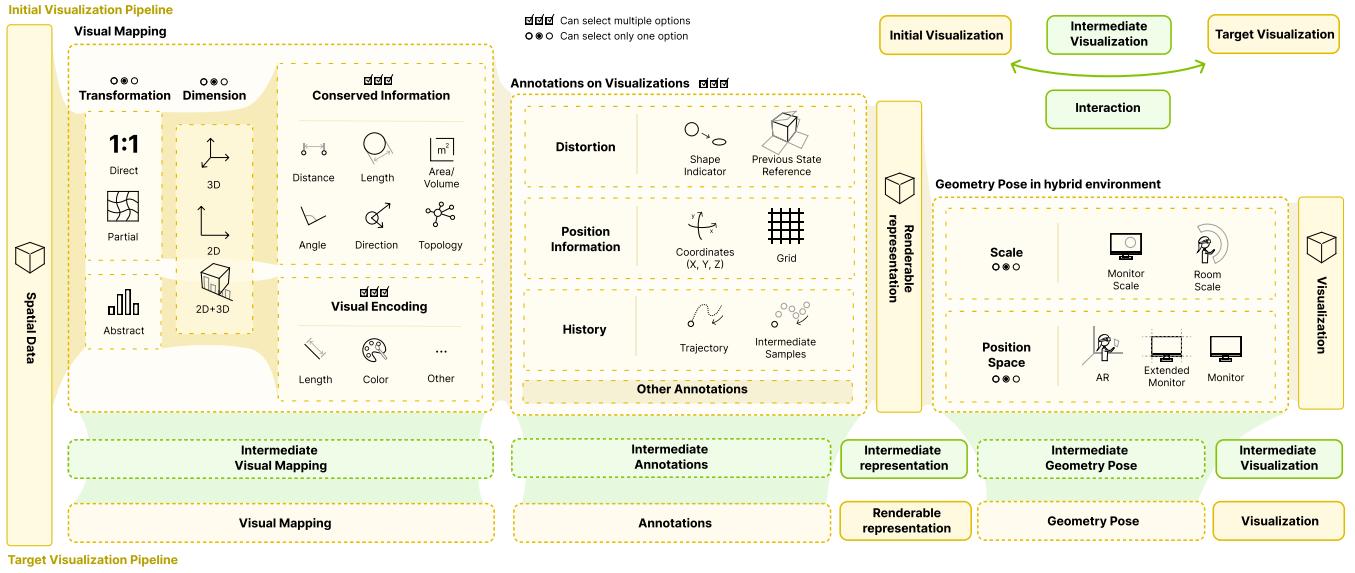
#### 4.1. Pipeline

We use a pipeline to describe how a visualization is presented, and therefore, how the spatial data undergoes several design choices before finally being presented as a visualization (Figure 2). Specifically, the **Visual Mapping** maps the data to the renderable visualization (Bruckner et al.'s [BIRW19] *visualization mapping* or a common visualization pipeline [CMS99]). The **Annotation** adds extra information to enhance the spatial understanding and the **Geometry Pose** discusses putting the renderable visualization in the actual environment (Bruckner et al.'s *output mapping*).

##### 4.1.1. Visual mapping

The visual mapping is the most important step of the visualization pipeline that consists of a set of transformations, which “repeatedly transform data into different forms and ultimately transform it into a representation that can be rendered by the computer system” [SM05]. This set of **transformations** may retain some of the intrinsic properties of 3D spatial datasets: geometry, topology, and attributes [SM05]. **Direct** transformations do not alter any of the intrinsic properties of the dataset. A molecule’s balls-and-sticks representation, e.g., does not change its properties throughout the transition. **Partial** transformations alter some but not all of the intrinsic properties and possibly alter the dimensionality of the representation. One example of a partial transformation is a map projection, which alters geometry and topology, but not necessarily attributes, e.g., landmass or elevation. Another example is to use a subset of the dataset, such as a slice of the volumetric data, or to use transparency [WLMW\*16] to hide a part of data and features. An **abstract** transformation does not retain any of the intrinsic properties of the spatial dataset, e.g., a plot of the total energy of a system. For example, Assor et al. [AMP\*24] transition between abstract bar chart presentation and concrete visualization that includes intrinsic physical properties. Another example is mapping the count of different proteins in a virus to a bar chart [SMR\*17].

The **dimension** impacts the way we display the visualization; it



**Figure 2:** Our design space for the visualization pipeline consists of three stages: The visual mapping, the additional annotations, and the geometry pose of the visualization embedded within the AR-desktop space. From left to right, the designer can make the design choices at each stage and create the final visualization. After several visualizations come out through this pipeline with slightly different design choices, we can determine the design for transitioning between them, as we illustrate in [Figure 3](#).

also later impacts the visual encoding considerations. A 2D representation may inherently partially discard positional information from the spatial dataset during the visual encoding. Depending on the chosen transformations, we consider the dimensionality of the resulting renderable data to be one of the following categories:

- **2D** visualizations are well suited for desktop PCs with a physical monitor setup. If the original data, however, is higher-dimensional (3D or nD), we first have to project it to two dimensions, such as via dimension reduction [VGdS\*20], losing some of the original data information. This loss of information can (partially) be compensated for by extra annotations or through interaction.
  - **3D** visualizations, in particular those of 3D spatial datasets, are naturally compatible with the AR environment. For 2D representations, extra information can be encoded on the extra dimension of the 3D representation. Time-based data can represent a specific slice of information at a given moment in 3D [MC00], although this approach does not capture the entire dataset.
  - **Hybrid 2D and 3D** combinations have the potential to reveal additional results [HHS\*24]. Therefore, such hybrid representations are well-suited for the AR-desktop hybrid environment. Ideally, the 2D part of the visualization is displayed on the monitor, and the 3D part in AR in a juxtaposed fashion [SFP\*23].
- It is important to determine which spatial information should be preserved (**conserved information**) to select the most appropriate visual mapping for a given dataset and application. For the actual implementation, it is also possible to pick several categories and mix them to retain part of each spatial attribute [NSZ\*17] or add customized constraints like parallel or self-intersection [MMG06]. A transformation that conserves the specified properties, however, does not necessarily exist for every possible dataset.
- **Distance, length** provides invariance of distance (isometry)—one of the most important geometric properties to provide accurate measurements between two points. While isometry is hard to

conserve within complex datasets with curved surfaces, it is still worth keeping the distance distortion to a minimum.

- **Area, volume** are commonly used in geographic maps [Sny87] as area-equal projections. When the distance is inevitably distorted, the invariance of the area will conserve scale information.
- **Angle, direction** mostly refer to conformal mapping [PPZ\*12], the visualization will keep its shape during a transition.
- **Topology** is commonly modified if the visualization is separated into several parts like in an exploding view [BG06] or a 3D model is expanded to a surface [KMM\*18, KFS\*17]. The loss of topology will impact the global structure information but potentially benefit local exploration.

**Visual encodings** are commonly used to represent data using visual variables by adjusting visual marks. For inherently spatial data, however, we need to be cautious about maintaining its spatial information. Visual encoding usually has to—at least partially—retain spatial information when, e.g., the original data is projected to a lower dimension. Thus, 3D or 2D spatial data is, most commonly, directly mapped using its positional information, with other visual channels such as color being used for additional data layers. This approach retains the original data semantics and aligns with the users' spatial and arrangement intuition. It also works well with AR environments due to the addition of stereoscopic vision and 3D input methods. If, in contrast, the **1:1** spatial mapping is replaced with **partially** spatial or with a non-spatial **abstract** one, such as to avoid occlusion or to have a quick overview by means of projection, distortion becomes an issue. The transition from a 1:1 globe representation to the Mercator projection, for instance, as illustrated in [Figure 5\(C\)](#), significantly distorts areas near the poles, but also provides a quick overview. The same effect also occurs when a user “returns” a visualization from the 3D AR space to the 2D monitor space due to the latter space's lower dimensionality.

#### 4.1.2. Geometry Pose

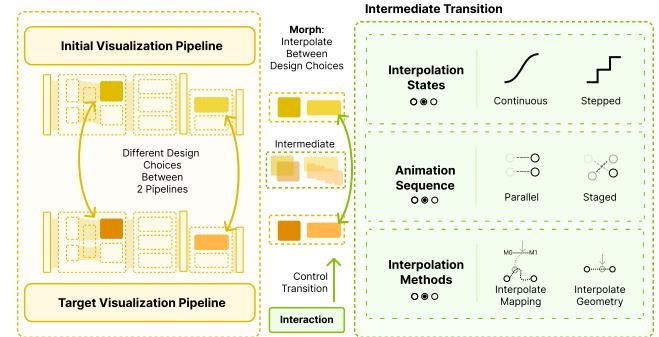
The **Geometry Pose** describes the scale and position of the visualization relative to the available scales and to the anchor positions in the AR-desktop environment setup. The first of its aspects, the **Scale**, impacts how the visualization is embedded in the environment [YJD\*18]. While previous work [CLK\*21] defined the scale based on the human size, we are (like others [ZIX\*24]) primarily concerned about the interaction with the visualization within the AR-desktop hybrid environment and thus classify it according to the visualization's size relationship as it is embedded in a space:

- **Monitor scale** describes a small scale of visualization that can be embedded anywhere, like users' hands (hand scale), physical monitors, or a desk (desk scale). This scale is the easiest to integrate into our proposed AR-desktop environment.
  - **Room scale** is large and cooperates well with AR space and creates the opportunity to inspect parts of the visualization in detail or get a completely new perspective (e.g., an inside-out view [YJD\*18]). In desktop settings, the monitor can no longer display the whole visualization due to its constrained screen size.
- In our case, a visualization's current position depends on the device that is displaying it, as well as its current visual mapping:
- The **AR space** renders a visualization usually stereoscopically and in 3D, but 2D visualizations are also acceptable.
  - The **monitor space** is usually constrained by a rectangular plane, although curved or alternatively shaped monitors are included. This space best suits 2D visualizations, while 3D visualizations can be displayed using, e.g., a perspective projection.
  - An **extended monitor space** reduces the constraint of the monitor. Therefore, it is possible to virtually extend the monitor to an almost endless plane or surface.

#### 4.1.3. Annotations

Additional information provided by annotations can help viewers to understand the data, e.g., by giving visual indicators of the current distortion in a static view or during a transition. Many forms of annotations are possible based on the design intention to further support users in connecting visual mappings during the transition by providing different visual cues. Figure 5(D) shows several example annotations that illustrate their usage, which we discuss in more detail in our exoplanet case study in Section 5. Below, we focus on the annotations that visualization designers annotate structural information, since it is the most crucial part for spatial data. Other common annotations, such as labels, can, of course, be used as well.

- **Distortion cues** are intended to address the issue that distortion is difficult to detect for a given visualization without visual aids, especially for common map projection techniques. A **shape indicator**, such as the Tissot indicator [MC01], can directly show the effects of distortion when it is compared to the original physical spatial meaning. A duplicate view of a **previous state reference** also enhances distortion awareness.
- **Position information** provides a relative reference for improving the locating quality because the direct visual mapping of position is likely inadequate to convey the visualization information. **Coordinate axes** and coordinate **grids** are common means to improve readability of positional mappings.
- A **history** can enable users to keep track of position information



**Figure 3:** The transition between two visualizations is primarily determined by the differences in design choices between the two visualization pipelines. For each distinct pair of design choices, designers can choose intermediate transition design options (right side) to define how to transition between them. The progression of these transitions is controlled through user interaction.

during the transition similar to motion lines [JR05, SSBG10]. Visualizing a **trajectory** is a common way to track a sequence of data samples. **Intermediate samples** is a similar method for tracking the transition in which a full visualization is displayed. It can be applied when a trajectory is not feasible.

The mentioned annotation methods provide a general (but not exhaustive) view of possibilities in this category. Annotations usually cooperate well with partial transformations of the 3D spatial data and thus help the user to be aware of distortion and avoid implicit bias from distorted visual properties like length or size [CMXF21]. Without a direct way to show the connection between the states, such as an animated transition (morph) between the visualizations, however, it remains mentally demanding to make the connection between the two visualizations.

## 4.2. Transitions

Once the design choices for the initial and target visualization have been made (or were given by an application domain), we need to specify the morph between them. For each morph, we focus on three factors: (1) the states of the interpolation, (2) the morph order of all visualization glyphs or elements, and (3) the interpolation method (see Figure 3). A common way to carry out a morph is to employ it as an animation. A well-designed animation can provide a smooth morph experience and result in a higher engagement and understanding for the user [HR07]. Still, it can also become time-consuming to need to wait until the animation ends [TMB02]. The morph, however, can also be used to explore intermediate states between the initial and final ones—if they are meaningful. Such a meaningful visualization, however, is usually carefully designed, and a direct interpolation of geometry toward or away from it can easily break the envisioned representation and become confusing.

The **interpolation states** is best designed based on the visual mapping. Many transformations are hard to understand or to explain well, even with adequate annotations. Separating animations that would normally happen simultaneously can provide substantial benefits for people to more easily understand the connection between the two states and sub-states [BB99].

- **Continuous** interpolations can provide viewers with a smooth morph of the visualization, which can be used for simple transformations that do not cause severe self-interaction or occlusion during the transforming procedure.
- **Stepped** interpolations have one or several independent steps during the animation. This method facilitates the separation of changes that otherwise occur simultaneously in the morph: it puts these changes into a sequential order. In the Mercator projection, e.g., the first step is to project the data to a cylinder, and the second step is to unfold the cylinder into a rectangle [Sny87]. We can also combine multiple states and present them as a cohesive stepped animation (e.g., Figure 6(d, e), where the animation consists of two steps: position transition and symbol transition).

The **animation sequence** focuses on how to animate each glyph or element of the visualization in order. Unlike the **interpolation states**, the animation sequence focuses on how to animate the whole collection of glyphs or elements in a specific time order.

- A **parallel** sequence transforms the whole visualization at the same time. It is common to use this mode for simple projections and data with meaningful intermediate states.
- A **staged** sequence starts with one or a few glyphs or elements and subsequently animates others. This approach can be responsive to user interaction, providing means to explore intermediate states interactively. For example, when the user “returns” a visualization and “pushes” it onto the PC screen, only those (3D) glyphs or elements of the visualization that begin to touch the screen surface start their animation procedure.

A straightforward interpolation of the element or glyph positions between the initial and target visualizations is possible for simple projections, but might disrupt the conserved information of the original visual mapping. To preserve the constraints of the visualization and the semantic connection between initial and final visualizations, the interpolation method may account for these factors and should be implemented with alternative algorithms. Ultimately, two major forms of **interpolation methods** are possible:

- The **interpolation of geometry** is a common and simple way to morph between two states by directly interpolating the final geometry positions of the initial and final visualizations. For each vertex of a geometry, its initial and final position is computed, and then the designed interpolation function between these two positions is applied. This method may distort the conversed information for the initial and target state, such as angles and area.
- The **interpolation of mapping** transitions between two visual mappings at the conceptual level, including the **conserved information** we had mentioned before. For instance, if the initial visualization conserves area while the final visualization conserves angles, the interpolation between both mappings will gradually reduce the preserved area concept and increase the preserved angle information concept of the visualization. When both visualizations share the same property, the mapping interpolation naturally keeps this property during the morph, offering an ideal continuum [PPZ<sup>\*</sup>12] within the intermediate states that potentially provide a meaningful representation. Another simple example is the unfolding of a cube. Linear interpolation would distort the edges of the cube, while the interpolation of the mapping constrains the length of the edges. Except for conserved information, we can also interpolate between different visual encodings, for which

we show an example in Figure 6. Here, the transition switches between the molecule glyph and chemical element symbols.

## 5. Case Studies

To illustrate the utility of our design space, we now discuss three case studies: an exoplanet dataset, medical MRI data, and chemical molecule data. We believe that these case studies from different application domains of scientific data analysis showcase a representative amount of design choices possible in our design space. We refer the reader to the accompanying video for a visual explanation.

### 5.1. Implementation

To implement the prototypes, we used Unity and the Mixed-Reality Toolkit (MRTK) to develop two different applications targeting the desktop platform (Windows) and AR headsets. We deployed the AR application on either the Oculus Quest 3 or the HoloLens 2, and ran the desktop component on a Windows desktop, together forming our AR-desktop environment. To set up this environment, we first calibrated the physical monitor by pointing to three of its corners in AR space with the index finger. We also established a local Wi-Fi network to synchronize user input and virtual object data between the AR and desktop platforms in real time.

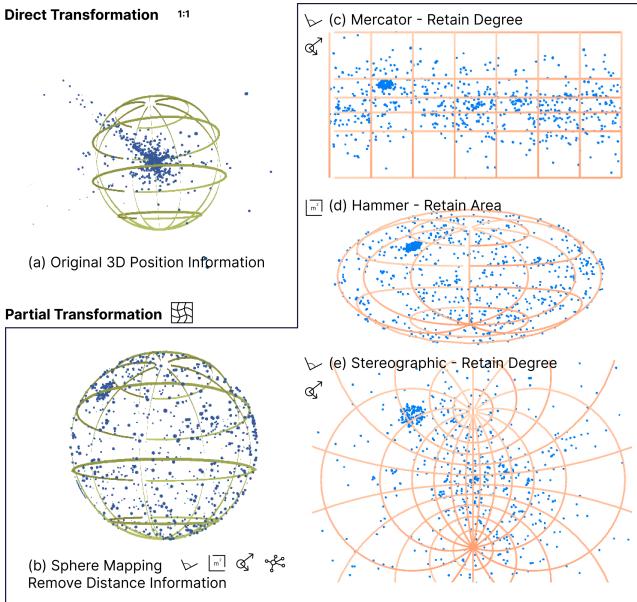
Our prototype incorporates a basic interaction mechanism that facilitates the transition across devices. Users can drag visualizations from desktop monitors into the AR space and vice versa. This drag operation can be performed using either hand gestures or a mouse. A typical interaction involves dragging a visualization off the desktop screen, triggering a transition from 2D to 3D visualization. Based on this implementation, we now describe the three case studies to illustrate our design space and showcase its descriptive power.

### 5.2. Exoplanet dataset

As a representative of sparse 3D point data, we selected NASA’s dataset of stellar systems with known exoplanets [Cal25], which records information about the respective stellar system and its planets. To describe the location of a system, experts use the International Celestial Reference System (ICRS) [MAE<sup>\*</sup>98]—a spherical coordinate similar to a geographic map. It is thus meaningful to use projection techniques similar to a terrestrial map, which in turn can be represented as a 3D sphere or projected onto a 2D plane using various established projection techniques. Here, we visualize primarily the locations of the respective stars, as opposed to the information about the planets (which are naturally much closer to their home star than the stars are separated from each other on average).

**Variation of design choices.** To map the data to the actual visualization, it is essential to define its appearance in 3D (AR) and 2D (screen) spaces. Figures 4 and 5 illustrate some of the visual mappings of the dataset we implemented as examples.

In 3D, all data points’ positional information can be **directly** mapped to the AR space with a **monitor scale** (Figure 5(A; d) and Figure 4(a))—essentially an *outside-in* view. Figure 5(B), in contrast, shows a **room size** view that simulates the sky—an *inside-out* view as if we would look outward from Earth. Alternatively, the data



**Figure 4:** Visual mapping examples for the exoplanet dataset. 3D: (a) direct mapping of the dataset with ICRS spherical coordinates as annotation; (b) location of the exoplanets orthogonally projected onto the ICRS surface. 2D: (c) Mercator, (d) Hammer, and (e) stereographic projection. All projections use a grid as an annotation.

can be represented as a sphere, sacrificing distance information (Figure 5(A; c) and Figure 4(b)). This approach can serve as a reference for visualizing the stars in the sky as we perceive them from Earth, again as an *outside-in view*. Such **partially** mapped positional data involves discarding some **distance** and **length** information during the mapping transition process.

When projecting the spherical view onto 2D to create an overview, distortion is inevitable. It is thus necessary to consider the retained spatial information after morphs. The 2D mappings in Figure 5(A) and Figure 4(d) use the Hammer projection that retains the **area** information. As another option, the interpolations in Figure 5(B) and (C) use Mercator and stereographic projections (also shown in Figure 4(c) and (e)). These two projections maintain **degree** and **direction**, a characteristic which is also called conformal mapping.

Even though choosing a given projection technique can selectively minimize the introduced distortion for a given application case, additional annotations can still be useful to help users identify this remaining distortion. Figure 5(D) illustrates a selection of common annotation examples that mitigate distortion issues in cartographic map projections and beyond. A **trajectory**, for instance, is frequently used to track the movement of single data points between two states to visualize the dynamics of the transition procedure. In addition, **coordinates** or **grids**, such as the Wulff net (also in Figure 5(D; a)), are commonly applied to visualize the space distortion. The Tissot indicator [MC01] (Figure 5(D; b)) is also a widely used **shape indicator** to further visualize the distortion of area and anisotropy in a map projection.

After finalizing the design decision for the initial and final visual mapping, the next part of designing a transition involves determining

the geometric pose. In most scenarios, we apply a **monitor scale** to the visualization, which offers viewers a convenient and seamless experience for transitioning the dataset using the mouse or gesture between **AR space** and **monitor space**. The extension of the space of a monitor or display is a common practical approach to enlarge the interactive surface [RD19], e.g., to improve the reach of mouse interaction. We refer to this setup as **extended monitor space**. This design accommodates the inherently 2D nature of traditional desktop input and screens [RD19]. It also avoids the depth issues and the limitation of 2 degrees of freedom (DOF) of input [ZFA22].

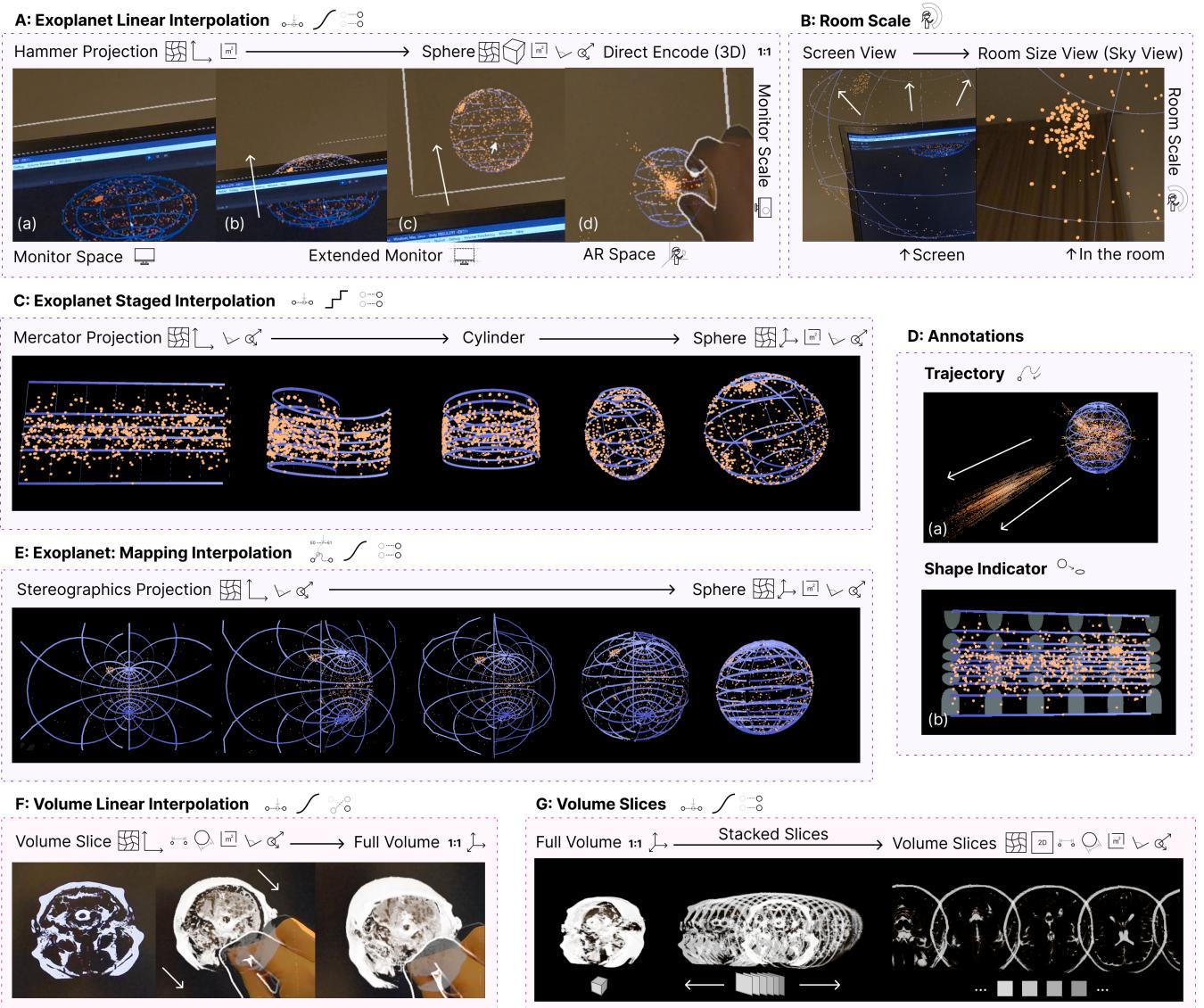
**Transition.** Based on the initial and target visualization of a transition as just defined, we can now construct several morphs between different design choices between two visualizations, for which Figure 5(A) illustrates a common example in our AR-desktop environment (also schematically in Figure 1(A)). Here, the dataset is initially mapped with the Hammer projection and embedded on the physical monitor. We then drag the visualization into 3D space and transition it to 3D spherical coordinates mapping. With mouse input, the visualization is dragged outside the **monitor space** and, during this transition, shown in (Figure 5(A; a–c)), simultaneously a transition and a morph happens here: The transition between design choice **monitor space**, **extended monitor space** and **AR space** as well as the morph between Hammer projection (keep **area** attribute), sphere projection (keep most geometry property except **distance**), and a **direct** mapping. For the animation between the different visualizations, we use **continuous** in **interpolation states**, **parallel** in **animation sequence**, and **interpolate geometry** in **interpolation methods**. This particular transition choice provides users with an uncomplicated solution for a smooth transition experience.

Further, Figure 5(C) and (E) illustrate another variation of the transition design. Figure 5(C) uses a **stepped** transition for the morph between Mercator projection and a 3D sphere to provide viewers with an intuitive explanation by first rolling the map to form a cylinder and then morphing this representation into a sphere. Figure 5(E) interpolates between the mapping of stereographic projection and spherical coordinates, which both keep invariant degree information. The interpolation thus also keeps the degree information during the transition and makes the whole transition conformal. The stepped character of this transition separates the whole transition into several semantical intermediate visualizations, in contrast to the continuous representation of the whole single transition. By interpolating the visual mapping rather than directly interpolating the geometric positions, more visual characteristics are retained, thus effectively minimizing extreme distortions during the transition.

### 5.3. Brain MRI data

The second data type in our case study is 3D volumetric data: data sampled on regular lattices and often displayed with direct volume rendering. Here, we use MRI data, which is widely used to analyze the internal structure of the human body in medicine [SST07] (other examples would be CT scans, fluid simulations, etc.). We specifically use the IEEE VIS 2010 contest dataset to demonstrate a volumetric data transition, as schematically shown in Figure 1(C).

We start with a common visualization mapping to explore the volume data's inner structure by utilizing a **partial** transformation,



**Figure 5:** Illustration of a series of transition examples from the exoplanet dataset (A–E) and the brain MRI dataset (F, G). In the figure we also show an example of a room sized representation (B) and of annotations to highlight distortion (D).

which shows individual slices. Figure 5(F) shows the process of transitioning from a 2D slice (**partial** transformation) to a **direct** (volume) representation, in which the slices are sequentially added as the user is ‘pulling the data out of the screen.’ Adding slices one by one, represents a **staged** animation sequence design choice. During this transition, we can still observe the intermediate visualization, where the visualization presents a subset of the original visualization—maintaining both its integrity and semantic meaning without any distortion of the spatial information. Hence, conserving **distance**, **length**, **angle**, **area**, and **direction**. This example demonstrates that the data’s spatial information is not manipulated—we merely show a continuously changing subset, which meaningfully retains a part of the original data and can still be treated as an integrated dataset for mapping designs. We can also transition the volume representation to a set of slice-based 2D representations as

we show in Figure 5(G), where each slice is one that was originally recorded in the MRI data. Here, we break the **topology** of the dataset and remove part of its global structure by animating the slices so that they end up being shown side-by-side as traditionally on a surgeon’s light box. The slices are fanned out along the x-axis, utilizing the **interpolate geometry** interpolation method and **parallel** animation sequence, while keeping the currently selected slice (on the desktop) in the center. This side-by-side representation does not introduce distortion within the slices, but does, for example, not conserve **distance** between the slices. For volumetric data, of course, it would also be feasible to extract isosurfaces and project them, similar to the mappings we used for the exoplanet dataset, through various design choices.

#### 5.4. Molecules

Finally, we present a use case of structured 3D data in the form of molecular diagrams. Graphical representations of molecules have a longstanding tradition in chemistry and biology, as clearly, not only the composition but also the relative spatial arrangement of the atoms determines the properties of a chemical compound. Pertinent to our discussion of the design space, the structure of molecules is not only expressed in 3D but also frequently and uniquely in 2D. In contrast to the first case study, here, the transformations are not described by a formula but rather by an algorithm that implements a defined set of rules [Bre08].

Traditional representations are known as structural or skeletal formulas (e.g., Figure 6(a)). They express which atoms are chemically bound to which other ones and may contain further qualifiers. Connecting lines often express the bond type (single, double) and may be related to details of the electron structure (in Lewis formulas, each line represents an electron pair). Although generally used as a 2D representation, they can qualitatively express spatial information via, e.g., wedges instead of lines. Traditional structural formulas thus primarily express molecule topology, but also provide—for the experienced chemist—a wealth of information about possible reactivity and properties. Naturally, a classical 2D projection of the 3D assembly onto a 2D screen cannot represent all the subtle details of a molecular conformation, and the spatial extent of certain parts of the molecule may not be obvious. Yet, this information is important for the stereochemical selectivity of reactions and the overall reactivity, so 3D views are used as well (e.g., Figure 6(f)). Moreover, for larger molecules, the mapping of 2D structural formulas and 3D spatial models may not be obvious—even for experienced chemists.

The molecule in our example (Figure 6) contains two double bonds in one of the side chains and another double bond in the other side chain. There is thus the possibility of an intramolecular reaction, which is quite obvious for chemists due to their familiarity with the representation. A reaction can only take place, however, if the spatial arrangement allows it, which is not visible in the 2D representation, and we thus transition the structural formula to its known 3D structure (Figure 6(f)). For this purpose, we apply spatial coordination during the morph from structure formula to a 3D ball-and-stick representation, i.e., we move the letter and line elements to their correct 3D positions as we transition. Yet, such a **continuous** interpolation (Figure 6(b, c)), when done on a desktop PC only, still requires interactive camera movement to fully grasp the alignment of the two side chains. When the morph is supported by a transition into the 3D/AR environment, in contrast, the chemist can fully concentrate on tracking the location of the double bonds. Their visibility and tracking can further be enhanced by a **stepped** interpolation (Figure 6(d, e)). Here, in the first step, we morph the atoms of the structure formula to their respective 3D positions and, afterward, smoothly blend into the 3D balls-and-sticks model. In both cases, we use a representation at **monitor scale**, but the **room scale** can be beneficial to investigate, e.g., datasets of crystals. Here, we chose the **position space** as **AR** because problems in chemistry often require a certain view angle on the molecules for occlusion-free vision on the region of interest, which requires interaction to rotate the molecule. The 3D manipulation using hand gestures is easier when done close to the user than next to the monitor.

A **staged** animation sequence is a possible alternative to enhance the spatial sense-making during the morph. For example, each side chain of the presented molecule could be animated in a staged fashion, which allows the chemist to concentrate only on one part of the molecule at a time. The trade-off here is that this would create several intermediate visualizations that are chemically “incorrect”. However, adding an animated **visual encoding** that reduces the opacity of the side chains that are currently not animated or in the final visualization could mitigate this effect. **Annotations** are either not possible to implement statically, like **distortion**, or would only clutter the visualization since the 3D visualization of molecular structures is already quite complex.

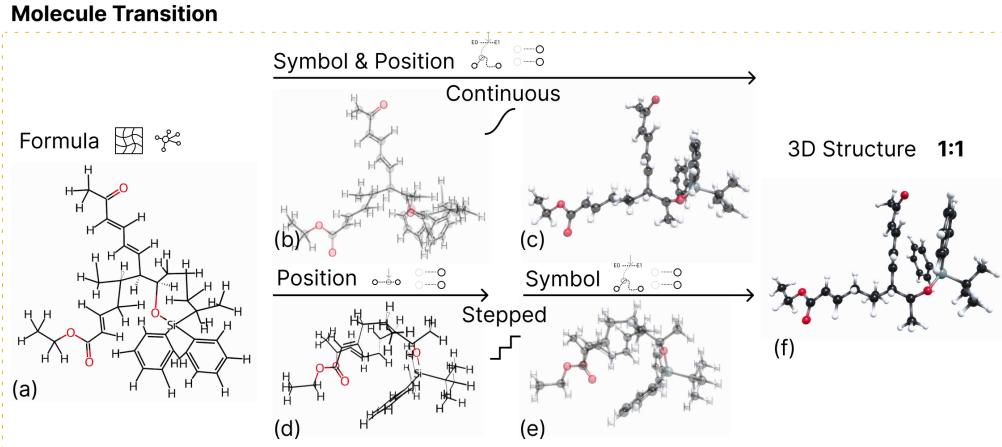
## 6. Discussion

While these case studies demonstrate the potential application of our design space, we also want to raise several points that affect its application and use in the future as well as its potential extension.

**Notion of scale.** In most cases, 3D spatial data inherently provides a literal scale of the data. Whether or not it is feasible to present the data on the literal scale, especially in cases where the literal scale is between monitor and room scale, is up for further investigation. In the examples we presented in our case studies, only the MRI dataset can be presented at its literal scale, which is, depending on the required analysis process, probably not the preferred choice by domain scientists. Therefore, the question of how to scale the data during the transition between environments strongly depends on the task at hand. Since the scale is difficult to generalize, we believe that giving users control over the size is the best solution.

**Composed visualizations.** Visualizations of 3D spatial data that are combined with 2D annotations or visualizations [HHS\*24] pose a whole new set of challenges in the context of AR-desktop hybrid environments. In a transition, for example, the 2D parts of the visualization could stay on the desktop PC, and only (annotation) lines could be provided to link between the representations and environments [SFP\*23]. But this process could also include a different set of transformation operations to reach a different final state of the 2D visualization. Therefore, we could separate the 2D and 3D parts of the visualization into two categories and regard the information separately during the animated transition. Our design space provides some options to separately process the visualizations, but would require further dimensions to provide comprehensive options for this kind of visualization.

**Intermediate states.** A transition design that emphasizes intermediate states by, e.g., using **interpolate mapping** together with a **staged** animation sequence or **stepped** interpolation states, provides valuable additional information that supports users in mentally connecting two representations. In our molecule use case, e.g., the intermediate state, in which the structure formula is arranged the same way as the balls-and-sticks visualization, provides a way to connect textbook knowledge with the spatial arrangement of the molecule. Some spatial datasets lend themselves well to intermediate states, but in general, they are quite difficult to create automatically, even with the utilization of our design space. Also, providing full control over the animation timeline (e.g., a scrollbar of a video) can be a powerful tool for data exploration using animated transitions.



**Figure 6:** Case study on molecular data. The atoms' positions are interpolated linearly between the 2D formula and the 3D structure. Upper animation path: **continuous** interpolation; lower animation path: **stepped** interpolation between the two visualizations.

**Interaction for transitions.** Our design space focuses on integrating spatial information and on the relationship between visualization and the display environment. While the original data contains the spatial information that is potentially sampled from reality, presenting the data within the AR-desktop space may also semantically assign real-world knowledge to the visualization or interaction. The area surrounding the monitor, for instance, can imply different types of transitions based on the direction relative to the monitor [RD19]. These spatial semantics can be stacked or juxtaposed to create complex layouts for more sophisticated designs. In addition, the position in the AR-desktop environment can be represented by three variables. Based on the positioning in the immersive environment, it is thus feasible to assign up to three parallel input variables. Each variable can correspond to a distinct state and can support multiple state changes simultaneously. The monitor's physical appearance is also usually rectangular, which can be used as a reference frame. As we focused on visual appearance in our work, however, we did not include such considerations as dedicated design dimensions.

**Multiple datasets.** While our case studies only come from three domains (yet with a representative range of different types of data), it is important to consider the broader context of data analytics systems. Each domain typically has unique spatial data types and specific tasks that are not easily generalized across all cases. It is also common for visual analytics systems to display multiple or combined visualizations simultaneously [KMM<sup>\*</sup>18]. A user may have different visualizations of the same dataset; each presenting distinct aspects of the data. We hypothesize that this concept can be applied to lay out multiple visualizations and, thus, open the possibility to transition between them. In addition, presenting the same information through different visualizations may help us reduce bias and enhance the user's overall understanding of the data.

**Animation.** We believe that the dimensions of our design space that influence the sequence of elements or glyphs have far more impact on a user's experience than animation curves or staggered animations. Nonetheless, being able to swap animation curves is a possible additional design dimension. Our approach inherently lets the user control animations while transitioning a visualization. We believe, however, that further control, like moving the object

in a direction that is orthogonal to the direction that triggers the transition, could possibly be used, or drag-and-drop operations could initiate dedicated animation sequences.

**Generalizability.** Describing transitions using our proposed design space is highly domain-specific. In general, it may be difficult to translate concepts that work on one domain's dataset to another domain. This fact also implies that some of our proposed design dimensions can be extended by considering further domains and datasets, especially for our annotation design dimensions. While our work focuses specifically on the AR-desktop environment, the visualization transition design within our chosen environment has the potential to extend to other cross-device settings, such as those involving tablets and AR headsets. When portable or multiple devices are involved, the geometric pose design space may further consider the spatial relationships among devices, the physical environment, and the visualizations. In addition to cross-device scenarios, transitions across different levels of virtuality—such as between VR, AR, and physical-world monitors [FAP<sup>\*</sup>22, AWK<sup>\*</sup>23]—also require considerations when designing transitions between different actualities.

**Hybrid input modalities.** For our prototypes, we developed a simple framework that focuses on AR-desktop environments. For a proper generalization of interaction across multiple devices [HMK<sup>\*</sup>19], there remains a need for a more universal framework to integrate different input devices while supporting smooth input transitions and seamless manipulation of virtual objects. Though existing studies [HWF<sup>\*</sup>22, SSP<sup>\*</sup>23, RIK<sup>\*</sup>25] debate whether users like to change input modalities, we believe that the traditional mouse and keyboard input will not be replaced anytime soon for desktop PCs [WBR<sup>\*</sup>20, PNB<sup>\*</sup>21]. For hybrid systems, however, developers should consider input modalities that can be used on all incorporated devices without the requirement of switching, and which then could be used to control the transitions we describe in our design space.

**User feedback.** In this work, we deliberately did not conduct an empirical experiment to validate our work, as our focus is the design space rather than evaluating the usability of a specific implemented instance of the interaction design or of given use cases. Our goal is to empower designers to describe, generate, and eval-

uate designs [BLBM21]. Given the domain-specificity of spatial dataset visualization tasks, a user study would have only offered limited general insight into the design space. Instead, we validated our approach through our diverse use case demonstrations in [Section 5](#). Moreover, prior studies have already examined the general effects of AR-desktop hybrid systems and demonstrated the user engagement in transferring virtual objects between screen and AR space [CMGS25, LCKP25, RIK\*25].

## 7. Conclusion

With our design space for transitions of spatial data between planar 2D displays and stereoscopic AR environments, we extended past work that looked at such transitions for non-spatial data [LCP\*22]. With our design space, we demonstrate how to make use of and cater to the unique spatial properties of data that arise in many scientific application domains such as medicine, astronomy, chemistry, physics, etc., for which we are not at liberty to use the third dimension in AR space to facilitate the transition between the different actualities. Instead, in our design space, we embrace this constraint and demonstrate how we can make use of different spatial mappings, annotations, and dedicated animation sequences to allow viewers to mentally follow the changing representation of the data as it transitions between the spaces. As we noted in our discussion, we do not make the claim that our design space would be complete. Domain-specific constraints and practices are likely to give rise to additional possibilities for designing transitions that, in turn, may then be applicable to other use cases. Nonetheless, we have laid the foundation for the discussion of how to design effective AR-desktop environments for the analysis of spatial data.

## Acknowledgments

We thank Ambre Assor for her feedback on our manuscript. This work is partially funded by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy—EXC 2075–390740016. We also acknowledge the support by the Stuttgart Center for Simulation Science (SimTech).

## Supplemental Material Pointers

We share our additional material (author version of the paper, self-created figures, video figure) at [osf.io/ve52m](https://osf.io/ve52m). We will also make the source code available on GitHub upon publication of the paper.

## Images/figures license/copyright

We as authors state that all of our figures in this article are and remain under our own personal copyright, with the permission to be used here. We also make them available under the [Creative Commons Attribution 4.0 International \(CC BY 4.0\)](#) license and share them at [osf.io/ve52m](https://osf.io/ve52m).

## References

- [AGF\*23] AUDA J., GRUENEFELD U., FALTAOUS S., MAYER S., SCHNEEGASS S.: A scoping survey on cross-reality systems. *ACM Comput Surv* 56, 4 (2023), 83:1–83:38. [doi:10/gtnn22](https://doi.org/10/gtnn22).
- [AMP\*24] ASSOR A., MCGUFFIN M. J., PROUZEAU A., DRAGICEVIC P., HACHET M.: Exploring the Design of Animated Transitions between Abstract and Concrete Visualizations in Immersive Environments. HAL preprint, 2024. [URL: hal.science/hal-04552817](https://hal.science/hal-04552817).
- [And74] ANDERSON P. S.: An oblique, poly-cylindrical, orthographic azimuthal equidistant cartographic projection: Its purpose, construction and theory. *Cartogr* 8, 4 (1974), 182–186. [doi:10/g8rvqx](https://doi.org/10/g8rvqx).
- [ARES13] ABUKHODAIR F. A., RIECKE B. E., ERHAN H. I., SHAW C. D.: Does interactive animation control improve exploratory data analysis of animated trend visualization? In *Proc. Electronic Imaging* (2013), SPIE, Bellingham, pp. 211–223. [doi:10/nvkc](https://doi.org/10/nvkc).
- [AWK\*23] AIGNER D., WANG N., KIELMAYER D., STEINER J., HOCH-PÖCHLER J., HEINZL C., ROTH D., MAURER F., ANTHES C.: Cardiac visualisation along the RV-continuum — A high-fidelity pilot study. In *Adjunct Proc. ISMAR* (2023), IEEE Comp. Soc., Los Alamitos, pp. 675–680. [doi:10/g84w4g](https://doi.org/10/g84w4g).
- [BB99] BEDERSON B. B., BOLTMAN A.: Does animation help users build mental maps of spatial information? In *Proc. InfoVis* (1999), IEEE Comp. Soc., Los Alamitos, pp. 28–35. [doi:10/btv9mv](https://doi.org/10/btv9mv).
- [BBK\*06] BORNIK A., BEICHEL R., KRUIJFF E., REITINGER B., SCHMALSTIEG D.: A hybrid user interface for manipulation of volumetric medical data. In *Proc. 3DUI* (2006), IEEE Comp. Soc., Los Alamitos, pp. 29–36. [doi:10/dpzb7z](https://doi.org/10/dpzb7z).
- [BG06] BRUCKNER S., GRÖLLER M. E.: Exploded views for volume data. *IEEE Trans Vis Comput Graph* 12, 5 (2006), 1077–1084. [doi:10/cg2hdq](https://doi.org/10/cg2hdq).
- [BGP\*11] BORKIN M., GAJOS K., PETERS A., MITSOURAS D., MELCHIONNA S., RYBICKI F., FELDMAN C., PFISTER H.: Evaluation of artery visualizations for heart disease diagnosis. *IEEE Trans Vis Comput Graph* 17, 12 (2011), 2479–2488. [doi:10/dnpmkq](https://doi.org/10/dnpmkq).
- [BIRW19] BRUCKNER S., ISENBERG T., ROPINSKI T., WIEBEL A.: A model of spatial directness in interactive visualization. *IEEE Trans Vis Comput Graph* 25, 8 (2019), 2514–2528. [doi:10/gjbdnn](https://doi.org/10/gjbdnn).
- [BKP01] BILLINGHURST M., KATO H., POUPYREV I.: The MagicBook: A transitional AR interface. *Comput Graph* 25, 5 (2001), 745–753. [doi:10/cz5nxn](https://doi.org/10/cz5nxn).
- [BLBM21] BEAUDOUIN-LAFON M., BØDKER S., MACKAY W. E.: Generative theories of interaction. *ACM Trans Comput-Hum Interact* 28, 6 (2021), 45:1–45:54. [doi:10/pqrk](https://doi.org/10/pqrk).
- [Bre08] BRECHER J.: Graphical representation standards for chemical structure diagrams (IUPAC recommendations 2008). *Pure Appl Chem* 80, 2 (2008), 277–410. [doi:10/cgx95](https://doi.org/10/cgx95).
- [BSB\*18] BACH B., SICAT R., BEYER J., CORDEIL M., PFISTER H.: The hologram in my hand: How effective is interactive exploration of 3D visualizations in immersive tangible augmented reality? *IEEE Trans Vis Comput Graph* 24, 1 (2018), 457–467. [doi:10/gcp924](https://doi.org/10/gcp924).
- [Cal25] CALIFORNIA INSTITUTE OF TECHNOLOGY: NASA exoplanet archive. Web site: [exoplanetarchive.ipac.caltech.edu](https://exoplanetarchive.ipac.caltech.edu), 2025. Last visited: Jun. 2025.
- [CDH\*19] CAVALLO M., DOLAKIA M., HAVLENA M., OCHELTREE K., PODLASECK M.: Immersive insights: A hybrid analytics system for collaborative exploratory data analysis. In *Proc. VRST* (2019), ACM, New York, pp. 9:1–9:12. [doi:10/gm8b84](https://doi.org/10/gm8b84).
- [CGS\*22] COOLS R., GOTTSACKER M., SIMEONE A., BRUDER G., WELCH G., FEINER S.: Towards a desktop-AR prototyping framework: Prototyping cross-reality between desktops and augmented reality. In *Adjunct Proc. ISMAR* (2022), IEEE Comp. Soc., Los Alamitos, pp. 175–182. [doi:10/gsghhv](https://doi.org/10/gsghhv).
- [CLK\*21] CHELIOTIS K., LIAROKAPIS F., KOKLA M., TOMAI E., PASTRA K., DARRA A., BEZERIANOU M., KAVOURAS M.: A classification of augmented reality approaches for spatial data visualization. In *Adjunct Proc. ISMAR* (2021), IEEE Comp. Soc., Los Alamitos, pp. 367–372. [doi:10/g8rvrg](https://doi.org/10/g8rvrg).

- [CMGS25] COOLS R., MAERVOET I., GOTTSACKER M., SIMEONE A. L.: Comparison of Cross-Reality Transition Techniques Between 3D and 2D Display Spaces in Desktop-AR Systems. *IEEE Transactions on Visualization and Computer Graphics* 31, 5 (2025), 2798–2805. [doi:10.1109/TVCG.2025.3549907](https://doi.org/10.1109/TVCG.2025.3549907).
- [CMS99] CARD S. K., MACKINLAY J. D., SHNEIDERMAN B.: *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann, San Francisco, 1999. [urn:oclc:record:1036805840](https://urn.nbn.se/resolve?urn=urn:nbn:se:mfki:record:1036805840).
- [CMXF21] CEJA C. R., MCCOLEMAN C. M., XIONG C., FRANCONERI S. L.: Truth or square: Aspect ratio biases recall of position encodings. *IEEE Trans Vis Comput Graph* 27, 2 (2021), 1054–1062. [doi:10.1109/TVCG.2020.300055m](https://doi.org/10.1109/TVCG.2020.300055m).
- [DBJ\*11] DRAGICEVIC P., BEZERIANOS A., JAVED W., ELMQVIST N., FEKETE J.-D.: Temporal distortion for animated transitions. In *Proc. CHI* (2011), ACM, New York, pp. 2009–2018. [doi:10.1109/CHI.2011.5770709](https://doi.org/10.1109/CHI.2011.5770709).
- [DE98] DIX A., ELLIS G.: Starting simple: Adding value to static visualisation through simple interaction. In *Proc. AVI* (1998), ACM, New York, pp. 124–134. [doi:10.1109/AVI.1998.66153](https://doi.org/10.1109/AVI.1998.66153).
- [DRST14] DÜBEL S., RÖHLIG M., SCHUMANN H., TRAPP M.: 2D and 3D presentation of spatial data: A systematic review. In *Proc. 3DV* (2014), IEEE Comp. Soc., Los Alamitos, pp. 11–18. [doi:10.1109/3DV.2014.68](https://doi.org/10.1109/3DV.2014.68).
- [EDF08] ELMQVIST N., DRAGICEVIC P., FEKETE J.-D.: Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *IEEE Transactions on Visualization and Computer Graphics* 14, 6 (2008), 1159–1148. [doi:10.1109/TVCG.2008.153](https://doi.org/10.1109/TVCG.2008.153).
- [FAP\*22] FRÖHLER B., ANTHES C., POINTECKER F., FRIEDL J., SCHWAJDA D., RIEGLER A., TRIPATHI S., HOLZMANN C., BRUNNER M., JODLBAUER H., JETTER H.-C., HEINZL C.: A survey on cross-virtuality analytics. *Comput Graph Forum* 41, 1 (2022), 465–494. [doi:10.1111/cgf.14202](https://doi.org/10.1111/cgf.14202).
- [FS91] FEINER S., SHAMASH A.: Hybrid user interfaces: Breeding virtually bigger interfaces for physically smaller computers. In *Proc. UIST* (1991), ACM, New York, pp. 9–17. [doi:10.1145/108053.108054](https://doi.org/10.1145/108053.108054).
- [GFM\*22] GALL A., FRÖHLER B., MAURER J., KASTNER J., HEINZL C.: Cross-virtuality analysis of rich X-ray computed tomography data for materials science applications. *Nondestr Test Eval* 37, 5 (2022), 566–581. [doi:10.1109/ndt.2022.3161001](https://doi.org/10.1109/ndt.2022.3161001).
- [GHFH23] GALL A., HEIM A., FRÖHLER B., HEINZL C.: Uncertainty unveiled: Revealing the uncertainty of distribution visualization through cross reality. In *Adjunct Proc. ISMAR* (2023), IEEE Comp. Soc., Los Alamitos, pp. 20–24. [doi:10.1109/ISMAR55505.2023.9790001](https://doi.org/10.1109/ISMAR55505.2023.9790001).
- [GHW\*24] HONG J., HNATYSHYN R., SANTOS E. A. D., MACIEJEWSKI R., ISENBERG T.: A survey of designs for combined 2D+3D visual representations. *IEEE Trans Vis Comput Graph* 30, 6 (2024), 2888–2902. [doi:10.1109/TVCG.2024.359751](https://doi.org/10.1109/TVCG.2024.359751).
- [HMK\*19] HORAK T., MATHISEN A., KLOKMOSE C. N., DACHSELT R., ELMQVIST N.: Vistribute: Distributing interactive visualizations in dynamic multi-device setups. In *Proc. CHI* (2019), ACM, New York, pp. 616:1–616:13. [doi:10.1145/3290609.3300710](https://doi.org/10.1145/3290609.3300710).
- [HMK\*20] HALLADJIAN S., MIAO H., KOUŘIL D., GRÖLLER M. E., VIOLA I., ISENBERG T.: Scale Trotter: Illustrative visual travels across negative scales. *IEEE Trans Vis Comput Graph* 26, 1 (2020), 654–664. [doi:10.1109/TVCG.2019.2890001](https://doi.org/10.1109/TVCG.2019.2890001).
- [HR07] HEER J., ROBERTSON G.: Animated transitions in statistical data graphics. *IEEE Trans Vis Comput Graph* 13, 6 (2007), 1240–1247. [doi:10.1109/TVCG.2007.33331](https://doi.org/10.1109/TVCG.2007.33331).
- [HRD\*19] HURTER C., RICHE N. H., DRUCKER S. M., CORDEIL M., ALLIGIER R., VUILLEMOT R.: FiberClay: Sculpting three dimensional trajectories to reveal structural insights. *IEEE Trans Vis Comput Graph* 25, 1 (2019), 704–714. [doi:10.1109/TVCG.2018.2870001](https://doi.org/10.1109/TVCG.2018.2870001).
- [HWF\*22] HUBENSCHMID S., WIELAND J., FINK D. I., BATCH A., ZAGERMANN J., ELMQVIST N., REITERER H.: ReLive: Bridging in-situ and ex-situ visual analytics for analyzing mixed reality user studies. In *Proc. CHI* (2022), ACM, New York, pp. 24:1–24:20. [doi:10.1145/3491102.3491111](https://doi.org/10.1145/3491102.3491111).
- [Jac05] JACKSON R. E.: Stereographic projection techniques for geologists and civil engineers, second edition: (richard j. isle and peter r. leyshon). *Environmental & Engineering Geoscience* 11, 2 (2005), 184–185. [doi:10.2113/gscg.11.2.184](https://doi.org/10.2113/gscg.11.2.184).
- [Jen12] JENNY B.: Adaptive composite map projections. *IEEE Trans Vis Comput Graph* 18, 12 (2012), 2575–2582. [doi:10.1109/TVCG.2012.264011](https://doi.org/10.1109/TVCG.2012.264011).
- [JR05] JOSHI A., RHEINGANS P.: Illustration-inspired techniques for visualizing time-varying data. In *Proc. VIS* (2005), IEEE Comp. Soc., Los Alamitos, pp. 679–686. [doi:10.1109/VIS.2005.1537011](https://doi.org/10.1109/VIS.2005.1537011).
- [KFS\*17] KRONE M., FRIESS F., SCHARNOWSKI K., REINA G., FADEMRECHT S., KULSCHEWSKI T., PLEISS J., ERTL T.: Molecular surface maps. *IEEE Trans Vis Comput Graph* 23, 1 (2017), 701–710. [doi:10.1109/TVCG.2016.2567979](https://doi.org/10.1109/TVCG.2016.2567979).
- [KMM\*18] KREISER J., MEUSCHKE M., MISTELBAUER G., PREIM B., ROPINSKI T.: A survey of flattening-based medical visualization techniques. *Comput Graph Forum* 37, 3 (2018), 597–624. [doi:10.1111/cgf.13690](https://doi.org/10.1111/cgf.13690).
- [LCKP25] LIAO S., CHAUDHRI S., KARWA M. K., POPESCU V.: SeamlessVR: Bridging the immersive to non-immersive visualization divide. *IEEE Trans Vis Comput Graph* 31, 5 (2025), 2806–2816. [doi:10.1109/TVCG.2025.3549925](https://doi.org/10.1109/TVCG.2025.3549925).
- [LCP\*22] LEE B., CORDEIL M., PROUZEAU A., JENNY B., DWYER T.: A design space for data visualisation transformations between 2D and 3D in mixed-reality environments. In *Proc. CHI* (2022), ACM, New York, pp. 25:1–25:14. [doi:10.1145/3491102.3491111](https://doi.org/10.1145/3491102.3491111).
- [LGV\*16] LAWONN K., GLASSER S., VILANOVA A., PREIM B., ISENBERG T.: Occlusion-free blood flow animation with wall thickness visualization. *IEEE Trans Vis Comput Graph* 22, 1 (2016), 728–737. [doi:10.1109/TVCG.2015.2474411](https://doi.org/10.1109/TVCG.2015.2474411).
- [LNP\*23] LU F., NANJAPPAN V., PARSONS P., YU L., LIANG H.-N.: Effect of display platforms on spatial knowledge acquisition and engagement: An evaluation with 3D geometry visualizations. *J Vis* 26, 3 (2023), 667–686. [doi:10.1111/jvis.13032](https://doi.org/10.1111/jvis.13032).
- [LSC\*23] LEE B., SATYANARAYAN A., CORDEIL M., PROUZEAU A., JENNY B., DWYER T.: Deimos: A grammar of dynamic embodied immersive visualisation morphs and transitions. In *Proc. CHI* (2023), ACM, New York, pp. 810:1–810:18. [doi:10.1145/3491102.3491111](https://doi.org/10.1145/3491102.3491111).
- [MAB\*18] MOHAMMED H., AL-AWAMI A. K., BEYER J., CALI C., MAGISTRETTI P., PFISTER H., HADWIGER M.: Abstractocyte: A visual tool for exploring nanoscale astroglial cells. *IEEE Trans Vis Comput Graph* 24, 1 (2018), 853–861. [doi:10.1109/TVCG.2017.2704666](https://doi.org/10.1109/TVCG.2017.2704666).
- [MAE\*98] MA C., ARIAS E. F., EUBANKS T. M., FEY A. L., GONTIER A.-M., JACOBS C. S., SOVERS O. J., ARCHINAL B. A., CHARLOT P.: The international celestial reference frame as realized by very long baseline interferometry. *Astron. J* 116, 1 (1998), 516–546. [doi:10.1086/300100](https://doi.org/10.1086/300100).
- [MC00] MA K.-L., CAMP D. M.: High performance visualization of time-varying volume data over a wide-area network. In *Proc. SC* (2000), IEEE Comp. Soc., Los Alamitos, pp. 29:1–29:10. [doi:10.1109/SC.2000.881300](https://doi.org/10.1109/SC.2000.881300).
- [MC01] MULCAHY K. A., CLARKE K. C.: Symbolization of map projection distortion: A review. *Cartogr Geogr Inf Sci* 28, 3 (2001), 167–182. [doi:10.1080/08888690109349466](https://doi.org/10.1080/08888690109349466).
- [MDLI\*18] MIAO H., DE LLANO E., ISENBERG T., GRÖLLER M. E., BARIŠIĆ I., VIOLA I.: DimSUM: Dimension and scale unifying map for visual abstraction of DNA origami structures. *Comput Graph Forum* 37, 3 (2018), 403–413. [doi:10.1109/TVCG.2018.2870001](https://doi.org/10.1109/TVCG.2018.2870001).
- [MJC23] McDADE J., JING A., CUNNINGHAM A.: *RealityDrop: A Multimodal Mixed Reality Framework to Manipulate Virtual Content between Cross-system Displays*. arXiv preprint 2310.07458, 2023. [doi:10.1109/TVCG.2023.3549923](https://doi.org/10.1109/TVCG.2023.3549923).

- [MMG06] MERRY B., MARAIS P., GAIN J.: Animation space: A truly linear framework for character animation. *ACM Trans Graph* 25, 4 (2006), 1400–1423. [doi:10/dx5whc](#).
- [MNT\*24] MÖRTH E., NIELSEN C., TURNER M., XIANHAO L., KELLER M., BEYER J., PFISTER H., CHEN Z.-T., GEHLENBORG N.: *A Mixed Reality and 2D Display Hybrid Approach for Visual Analysis of 3D Tissue Maps*. OSF preprint, 2024. [doi:10/pqfd](#).
- [MTUK95] MILGRAM P., TAKEMURA H., UTSUMI A., KISHINO F.: Augmented reality: A class of displays on the reality-virtuality continuum. In *Proc. Telemanipulator and Telepresence Technologies* (1995), SPIE, Bellingham, pp. 282–292. [doi:10/dh8jnv](#).
- [MVB\*17] MEUSCHKE M., VOSS S., BEUING O., PREIM B., LAWONN K.: Combined visualization of vessel deformation and hemodynamics in cerebral aneurysms. *IEEE Trans Vis Comput Graph* 23, 1 (2017), 761–770. [doi:10/f92d8h](#).
- [NSZ\*17] NADEEM S., SU Z., ZENG W., KAUFMAN A., GU X.: Spherical parameterization balancing angle and area distortions. *IEEE Trans Vis Comput Graph* 23, 6 (2017), 1663–1676. [doi:10/f97b8n](#).
- [NVV\*06] NERI E., VANNOZZI F., VAGLI P., BARDINE A., BAR-TOLOZZI C.: Time efficiency of CT colonography: 2D vs 3D visualization. *Comput Med Imaging Graph* 30, 3 (2006), 175–180. [doi:10/br3jwx](#).
- [PFKJA24] POINTECKER F., FRIEDL-KNIRSCH J., JETTER H.-C., ANTHES C.: From real to virtual: Exploring replica-enhanced environment transitions along the reality-virtuality continuum. In *Proc. CHI* (2024), ACM, New York, pp. 799:1–799:13. [doi:10/g8rvq9](#).
- [PNB\*21] PAVANATTO L., NORTH C., BOWMAN D. A., BADEA C., STOAKLEY R.: Do we still need physical monitors? An evaluation of the usability of AR virtual monitors for productivity work. In *Proc. VR* (2021), IEEE Comp. Soc., Los Alamitos, pp. 759–767. [doi:10/gj3dh5](#).
- [PPZ\*12] PETKOV K., PAPADOPOULOS C., ZHANG M., KAUFMAN A. E., GU X.: Interactive visibility retargeting in VR using conformal visualization. *IEEE Trans Vis Comput Graph* 18, 7 (2012), 1027–1040. [doi:10/d9m2j9](#).
- [RD19] REIPSCHLÄGER P., DACHSELT R.: DesignAR: Immersive 3D-modeling combining augmented reality with interactive displays. In *Proc. ISS* (2019), ACM, New York, pp. 29–41. [doi:10/gtdktf](#).
- [RIK\*25] RAU T., ISENBERG T., KOEHN A., SEDLMAIR M., LEE B.: Traversing dual realities: Investigating techniques for transitioning digital objects between desktop and augmented reality environments. In *Proc. CHI* (2025), ACM, New York, pp. 1236:1–1236:16. [doi:10/pj4f](#).
- [RKP\*24] RAU T., KOCH M., PATHMANATHAN N., ISENBERG T., WEISKOPF D., SEDLMAIR M., KÖHN A., KURZHALS K.: Understanding collaborative learning of molecular structures in AR with eye tracking. *IEEE Comput Graph Appl* 44, 6 (2024), 39–51. [doi:10/nt8b](#).
- [RSK24] RAU T., SEDLMAIR M., KÖHN A.: chARpack: The chemistry augmented reality package. *J Chem Inf Model* 64, 12 (2024), 4700–4708. [doi:10/nt62](#).
- [SFP\*23] SCHWAJDA D., FRIEDL J., POINTECKER F., JETTER H.-C., ANTHES C.: Transforming graph data visualisations from 2D displays into augmented reality 3D space: A quantitative study. *Front Virtual Real* 4 (2023), 1155628:1–1155628:19. [doi:10/kvj9](#).
- [SM05] SCHROEDER W. J., MARTIN K. M.: Overview of visualization. In *Visualization Handbook*, Hansen C. D., Johnson C. R., (Eds.). Butterworth-Heinemann, Burlington, 2005, pp. 3–35. [doi:10/bkdf8t](#).
- [SMR\*17] SORGER J., MINDEK P., RAUTEK P., GRÖLLER E., JOHNSON G., VIOLA I.: Metamorphers: Storytelling templates for illustrative animated transitions in molecular visualization. In *Proc. SCCG* (2017), ACM, New York, pp. 2:1–2:10. [doi:10/gtmvsx](#).
- [Sny87] SNYDER J. P.: *Map Projections: A Working Manual*. Tech. Rep. 1395, U.S. Government Printing Office, 1987. [doi:10/gk8jp7](#).
- [SPZS24] SERAJI M. R., PIRAY P., ZAHEDNEJAD V., STUERZLINGER W.: Analyzing user behaviour patterns in a cross-virtuality immersive analytics system. *IEEE Trans Vis Comput Graph* 30, 5 (2024), 2613–2623. [doi:10/g8rvrd](#).
- [SSBG10] SCHMID J., SUMNER R. W., BOWLES H., GROSS M.: Programmable motion effects. *ACM Trans Graph* 29, 4 (2010), 57:1–57:9. [doi:10/dkrmpf](#).
- [SSP\*23] SCHRÖDER J.-H., SCHACHT D., PEPER N., HAMURCULU A. M., JETTER H.-C.: Collaborating across realities: Analytical lenses for understanding dyadic collaboration in transitional interfaces. In *Proc. CHI* (2023), ACM, New York, pp. 97:1–97:16. [doi:10/g8rvrc](#).
- [SST07] SCHULTZ T., SEIDEL H.-P., THEISEL H.: Topological visualization of brain diffusion MRI data. *IEEE Trans Vis Comput Graph* 13, 6 (2007), 1496–1503. [doi:10/cghw82](#).
- [TKAM06] TORY M., KIRKPATRICK A. E., ATKINS M. S., MÖLLER T.: Visualization task performance with 2D, 3D, and combination displays. *IEEE Trans Vis Comput Graph* 12, 1 (2006), 2–13. [doi:10/ccjthz](#).
- [TMB02] TVERSKY B., MORRISON J. B., BETRANCOURT M.: Animation: Can it facilitate? *Int J Hum Comput Stud* 57, 4 (2002), 247–262. [doi:10/d5frg6](#).
- [VGdS\*20] VERNIER E. F., GARCIA R., DA SILVA I. P., COMBA J. L. D., TELEA A. C.: Quantitative evaluation of time-dependent multi-dimensional projection techniques. *Comput Graph Forum* 39, 3 (2020), 241–252. [doi:10/g8r7s4](#).
- [WBR\*20] WANG X., BESANCON L., ROUSSEAU D., SERENO M., AMMI M., ISENBERG T.: Towards an understanding of augmented reality extensions for existing 3D data analysis tools. In *Proc. CHI* (2020), ACM, New York, pp. 528:1–528:13. [doi:10/gm4jj8](#).
- [WLMW\*16] WALDIN N., LE MUZIC M., WALDNER M., GRÖLLER E., GOODSELL D., LUDOVIC A., VIOLA I.: Chameleon: Dynamic color mapping for multi-scale structural biology models. In *Proc. VCBM* (2016), EG Assoc., Goslar, pp. 11–20. [doi:10/pqff](#).
- [WM22] WANG N., MAURER F.: A design space for single-user cross-reality applications. In *Proc. AVI* (2022), ACM, New York, pp. 31:1–31:5. [doi:10/g8rvrf](#).
- [YJD\*18] YANG Y., JENNY B., DWYER T., MARRIOTT K., CHEN H., CORDEIL M.: Maps and globes in virtual reality. *Comput Graph Forum* 37, 3 (2018), 427–438. [doi:10/gf78xn](#).
- [ZFA22] ZHOU Q., FITZMAURICE G., ANDERSON F.: In-depth mouse: Integrating desktop mouse into virtual reality. In *Proc. CHI* (2022), ACM, New York, pp. 354:1–354:17. [doi:10/g64wht](#).
- [ZHB\*22] ZAGERMANN J., HUBENSCHMID S., BALESTRUCCI P., FEUCHTNER T., MAYER S., ERNST M. O., SCHMIDT A., REITERER H.: Complementary interfaces for visual computing. *it – Inf Technol* 64, 4–5 (2022), 145–154. [doi:10/gtdkwp](#).
- [ZIX\*24] ZHAO L., ISENBERG T., XIE F., LIANG H.-N., YU L.: MeTACAST: Target- and context-aware spatial selection in VR. *IEEE Trans Vis Comput Graph* 30, 1 (2024), 480–494. [doi:10/gtnn25](#).