International Journal of Human-Computer Studies 118 (2018) 1–13

Contents lists available at ScienceDirect

International Journal of Human-Computer Studies

journal homepage: www.elsevier.com/locate/ijhcs

Assessing the usability of tile-based interfaces to visually navigate 3-D parameter domains

Daniel S. Lopes[a,b,*], Rafael K. dos Anjos[a,b,c], Joaquim A. Jorge[a,b]

[a] INESC-ID Lisboa, Portugal
[b] Instituto Superior Técnico, Universidade de Lisboa, Portugal
[c] FCSH, Universidade Nova de Lisboa, Portugal

A R T I C L E I N F O

Keywords:
Sliders
Rhombille tiling
HSV color space
Super-shape curves
3-D rotation

A B S T R A C T

Navigating 3-D parameter domains, such as color and orientation of an object, is a common task performed in most computer graphics applications. Although 1-D sliders are the most common interface for browsing such domains, they provide a tedious and difficult user experience that hampers finding desirable visual solutions. We present the Rhomb-i slider, a novel and visually enriching tile-based interface to navigate arbitrary 3-D parameter domains. Contrary to 1-D sliders, the Rhomb-i slider supports a sketch-based interface that gives simultaneous access to up to two parameters. We conducted a usability study to ascertain whether the proposed Rhomb-i slider is a more natural interface compared to 1-D sliders and other commonly used widgets for different 3-D parameter domains: HSV color space, super-shape curves, and rotation of a 3-D object. On the one hand, qualitative feedback and performance measures reveal that Rhomb-i sliders have similar results when compared to conventional HSV color interfaces, and are the preferred interface to efficiently explore the super-shapes parameter domain. On the other hand, Rhomb-i revealed to be a less efficient and effective interface to rotate a 3D object, thus paving the way to new design explorations regarding this tile-based interface.

1. Introduction

Visual navigation of parameter domains[1] requires interaction controls to adjust multiple parameters. Such visual tasks are commonly performed on many useful three-dimensional parameter domains such as color spaces, scale, translation and orientations of 3-D objects including shapes, lights and cameras. Even though 3-D parameter domains are not affected by the curse of dimensionality (Bellman, 2003), designing proper interaction components (Fekete, 2004) is still challenging since it requires mapping three-dimensional objects to flat two-dimensional media. Due to this dimensional gap, navigating and selecting a desirable visual goal within a 3-D parameter domain can be a tedious task as it usually is performed by trial-and-error probing followed by fine parameter tuning. To make matters worse, the vast expanse of a parameter domain contains uninteresting design solutions, requiring users to perform tiresome probing tasks to get at the desired optimal solution.

To tackle this dimensionality gap, users require appropriate graphical interfaces, preferably those which rapidly expand the access to large or compiled amounts of data, and can showcase desirable visual goals within a 3-D parameter domain.

However, the most commonly used interaction component are 1-D sliders as they can numerically represent any parameter type (Fekete, 2004). The most obvious limitation is that 1-D sliders can only explore a single path per interaction. This forces users to shift visual focus between the parameter domain interface and the content. This back-and-forth refocusing can be considered as a distraction since it requires constant attention shifts during navigation including parameter selection, adjustment and content analysis tasks.

Beyond 1-D sliders, different tailor-made visual interfaces have been developed to aid in exploration and selection tasks (Reisman et al., 2009, Yu et al., 2010; Jankowski and Hachet, 2013; Isenberg, 2016) for each 3-D parameter domain. Such specialized visual interfaces usually provide a wider and more flexible spectrum of visual solutions per interaction. Regardless of this advantage, each interface is customized for a specific parameter domain, leading to a wide variety of specialized widgets. This forces users to learn many different graphical interfaces, each with its own advantages, quirks and drawbacks.

To properly navigate 3-D parameter domains on flat media, it is thus necessary to translate the domain into a visible form that maps 3-D

* Corresponding author at: INESC-ID Lisboa, IST Taguspark, Room 2N9.1, Avenida Professor Cavaco Silva, Porto Salvo 2744-016 Portugal.
E-mail address: daniel.lopes@inesc-id.pt (D.S. Lopes).

[1] While *domain* is used interchangeably with *space* in this paper, we use the term to signify either discrete or continuous spaces defined by Cartesian or other coordinate (or parameter) system

https://doi.org/10.1016/j.ijhcs.2018.05.005
Received 9 December 2016; Received in revised form 15 May 2018; Accepted 17 May 2018
Available online 19 May 2018
1071-5819/

parametric solutions onto a bi-variate layout. Inspired by the Cubist Movement where 3-D objects are depicted from multiple viewpoints to be represented in a larger context, we propose a more suitable graphical representation for navigating 3-D parameter domains using a generic tile-based interface (please refer to Section 3.1 for more details). Due to their two-dimensional nature, tiles depict two parameters (or, equivalently, a pair of 1-D sliders) in a single graphical element. Hence, we can conceptualize a tile as a 2-D slider. Furthermore, by considering tiles with a diamond shape (i.e., rhombus form) and arranging three of them in a rhombille tiling pattern, we obtain an interface that can generally support any type of 3-D parameter domain. In order for this interface to allow simultaneously exploring multiple visual solutions per interaction, each tile should concurrently highlight important features inside the 3-D parameter domain. Highlighting these salient features can be accomplished by either using thumbnails, tags, figurines or other visual aids.

In this work, we propose and assess the usability of a rhombille shaped tile-based interface to visually navigate parameter domains with three dimensions. We wish to verify whether such tile-based interface makes for (1) a less time-consuming approach for users to search a 3-D parameter domain as compared to 1-D sliders and specialized widgets; and (2) a more general design model that can unify specialized widgets built for 3-D visual navigation and exploration.

The major contribution of this work is the Rhomb-i slider, a novel and visually rich tile-based interface for real-time browsing and selection within an arbitrary 3-D parameter domain. Since this is a new design, our work also describes the characteristics that make parameter domains more suitable to the Rhomb-i design model. To this end, we consider a hierarchy of parameter domains structured according to the complexity of visual objects, more specifically and in order of increasing complexity: colors, 2-D shapes and 3-D orientations of objects. We also conducted a user study to evaluate whether the proposed tile-based interface supports a more natural probing of three different domains, namely HSV color, super-shape curves, and 3-D rotations. Task completion time and participant preferences (or perceived satisfaction) were measured to validate the initial concept of the Rhomb-i slider.

2. Related work

In order to improve traditional sets of 1-D and 2-D sliders, several authors either proposed alternative interaction design techniques of conventional 1-D sliders (Perin and Dragicevic, 2014; Damaraju et al., 2013) or deconstructed the linear organization found in 1-D (Elmqvist et al., 2008; Satou et al., 2003; Schneider, 2010; Tsandilas et al., 2015; Eick, 1994; Barrios, 2014) and 2-D sliders (Johansson et al., 2014) into different spatial arrangements or considered tile-based interfaces (Bach et al., 2016; Rekimoto et al., 2001) to better access and perceive either multi-dimensional data or parameter domains (Fig. 1). Moreover, less conventional 2-D slider interfaces have even been implemented in commercial software (Tedeschi, 2014).

2.1. 1-D sliders

Regarding alternative 1-D slider interaction, Perin and Dragicevic (2014) presented a crossing-based widget of numerical table cells that leveraged constrained crossing gestures performed on a vertically stacked set of 1-D sliders. This promotes alternative interface styles and allows for a more efficient navigation when compared to standard widgets, as it enables simultaneous modification of multiple widgets similar to sliders. Damaraju et al. (2013) also addressed 1-D slider interaction and designed Multi-Tap sliders for prolonged multi-touch interaction. That interface was designed for quickly selecting and adjusting multiple numerical parameters, while users were encouraged to focus on the visual content rather than on the interface. Although using Multi-Tap offers several improvements over traditional 1-D sliders, the technique requires a 5–10 minute training phase and forces using multiple fingers with frequent repetitions of selection and parameter adjustments, making the experience rather choreographic.

As for alternative 1-D slider spatial arrangements, other authors presented various non-parallel slider configurations. Elmqvist et al. (2008) metaphorically considered a 2-D interactive visual canvas as a meadow where users can create and link starplots that display multivariate axes that function as sliders. Although the results indicate that the proposed technique provided a useful way of thinking and interacting with multivariate data, the complexity of the interface can easily scale up and it only supports comparisons between two or more datasets rather than interactive visualization of a single 3-D data set.

More interesting spatial arrangements were proposed to navigate parameter domains (Satou et al., 2003; Schneider, 2010; Tsandilas et al., 2015; Eick, 1994; Barrios, 2014). In particular, Tsandilas et al. (2015) presented SketchSliders that allow users to directly sketch 1-D sliders that can adopt arbitrary shapes with a pen stroke appearance which can be augmented with visual aids (e.g., markers, slider cursors, thumbnails of data distribution) (Eick, 1994), an approach that also can be used to explore multi-dimensional datasets. An unusual visual model was proposed by Barrios (2014) consisting of a hyper-cube where each vertex corresponds to a design solution. This is seemingly a logical arrangement represented with a multi-dimensional structure, that promotes a more efficient navigation since it displays several possible variations of parameters at once. However, the final result can easily originate complex layouts and become cluttered with too many candidate solutions.

Although these methods aim to improve slider interaction (Perin and Dragicevic, 2014; Damaraju et al.,2013) or introduce different slider spatial arrangements (Elmqvist et al., 2008; Satou et al., 2003; Schneider, 2010; Tsandilas et al., 2015; Barrios, 2014), they are applied to navigating multi-variate data sets or numerical tables and not continuous 3-D parameter domains. More importantly, the methods so far discussed are not real 2-D sliders in the sense that they do not allow controlling two parameters simultaneously.

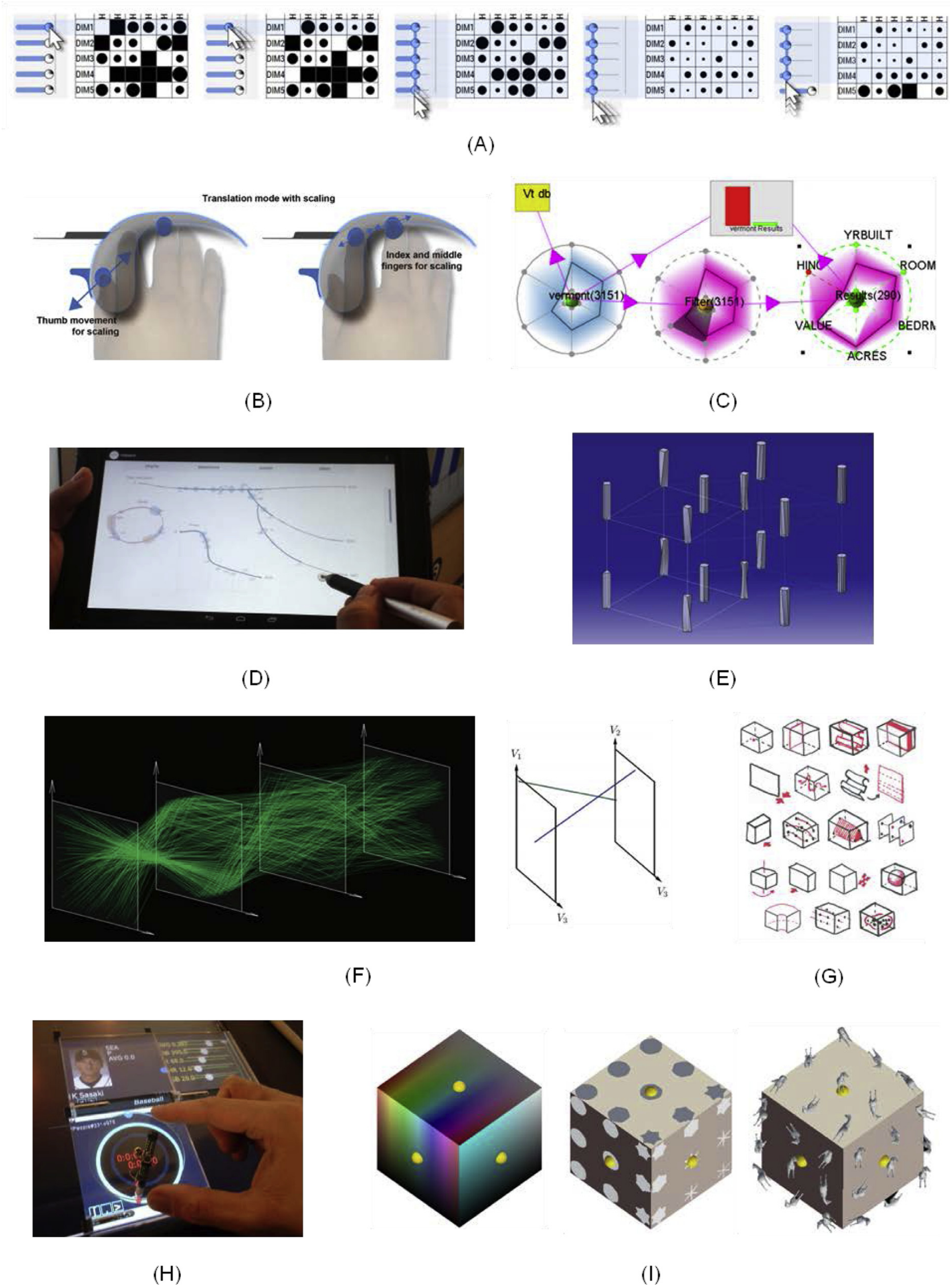
2.2. 2-D sliders

Given its bi-dimensional nature, several authors have considered 2-D sliders as a good visual encoding of parameter domains or data sets. To wit, Johansson et al. (2014) evaluated how well parallel coordinates can extend in a three-dimensional scene (parallel planes), and compared them to the standard two-dimensional parallel coordinates (parallel axes). Their evaluation detailed the usefulness of three-dimensional parallel coordinates for different tasks, along with the influence of axis configurations on identifying two-dimensional relationships (i.e., patterns) between variables in multivariate data. Interestingly, the study shows that adding a third dimension for analyzing 2-D patterns in parallel coordinates does not greatly benefit identifying patterns, in terms of either time or accuracy. The major limitations of the study were the focus on pattern recognition in a flat medium and that authors only considered a single axis configuration for each display, i.e., either parallel axes or parallel planes.[2]

2.3. Tile-based sliders

Tile-shaped graphical elements also have been used in the form of a cube to visualize temporal datasets, which appear frequently in many visual design scenarios. However, it is usually difficult to visualize space-time data, especially datasets that involve more than one spatial dimension, such as video and geo-temporal data. To address the visual design issues associated to these data, Bach et al. (2016) considered cubes as a suitable graphical element for temporal data visualization,

[2] http://papers.cumincad.org



(caption on next page)

also called as space-time cubes. The explicit spatial metaphor converts a cube into an easily-readable two-dimensional visualization of three-dimensional space-time data. However, these tile-shaped graphical elements are used for visualization purposes only and do not directly support interaction.

A literal approach to navigating parameter domains considered physical tiles as both graphical and tangible user interfaces (Rekimoto et al., 2001). Different transparent tiles are used to display, edit or manipulate data either independently or combined into more complex configurations. A specific type of tile presents grooves and is used for

Fig. 1. Alternative interface solutions for 1-D sliders, 2-D sliders and tile-based sliders: (A) Crossets - widgets based on crossing gestures for manipulating multiple aligned sliders simultaneously (Perin and Dragicevic, 2014); (B) Multi-tap Sliders - widget for multi-touch selection and adjustment of multiple parameters (Damaraju et al., 2013); (C) DataMeadow - canvas for constructing visual queries using starplots of multivariate data sets (Elmqvist et al., 2008); (D) SketchSliders - complex range sliders built using a sketch-based interface (Tsandilas et al., 2015); (E) Hyper-Matrix - visual model that showcases logical arrangements of parametric models (Barrios, 2014); (F) 3D parallel coordinates (Johansson et al., 2014); (G) generalized space-time cubes (Bach et al., 2016); (H) DataTiles - tangible tiles with graphical interfaces (Rekimoto et al., 2001); and (I) Rhomb-i sliders: proposed tile-based interface for HSV color, super-shape curves and 3-D rotation parameter domains. {(A) adapted from (Perin and Dragicevic, 2014) with author's permission; (B) adapted from with author's permission (Sriranga and Raju, 2013); (C) adapted from Elmqvist et al. (2008) with author's permission; (D) adapted from Tsandilas et al. (2015) with author's permission; (E) open access material from CumInCAD; (F) adapted from Johansson et al. (2014) with author's permission (G) adapted from Bach et al. (2016) with John Wiley and Sons permission; (H) adapted from Rekimoto et al. (2001) with author's permission}.

controlling other, content supporting, tiles. However, each groove functions as a regular 1-D slider and this method only considered rectangular tiles combined in grid-based compositions.

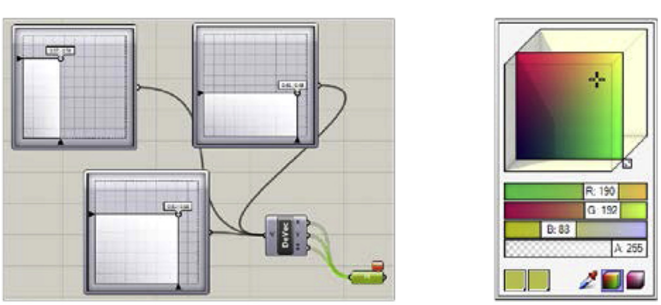
2.4. Commercial tools

Commercial software solutions have made use of 2-D sliders or square shaped tiles. Besides providing visual encodings for parameter domain visualization, a single square shaped tile also supports 2-D navigation and selection. Two distinctive tile-based interfaces that are worth mentioning were designed for parametric modeling using Grasshopper's visual programming language (Tedeschi, 2014). This software package presents a widget that aids in 3-D parameter domain navigation called MD Slider (multi-dimensional slider) consisting of a input surface represented as a quadrangular patch parameterized along two directions u and v (Fig. 2(A)). To access more than two parameters it is necessary to connect several input surface patches to other components within the process flow chart. Despite giving access to multi-dimensional parameters, the input surfaces are placed separately or, worse, overlapped within the working canvas. Furthermore, no visual feedback (e.g., glyphs, icons, color gradients) is rendered within the input surface besides the handle.

Similarly to the MD Slider, Grasshopper's Material Picker also presents a quadrangular input surface that the user can translate along the rail of a obliquely oriented one-dimensional slider (Fig. 2(B)). Contrary to the MD Slider, it only supports three parameters, and considering a single patch deeply limits the exploration of 3-D parameter domains as one of the dimensions is visually reduced to a point on the slider.

3. The Rhomb-i slider

The disadvantages of existing methods (Perin and Dragicevic, 2014; Damaraju et al., al., Sriranga and Raju, 2013; Elmqvist et al., 2008; Satou et al., 2003; Schneider, 2010; Tsandilas et al., 2015; Eick, 1994; Barrios, 2014; Johansson et al., 2014; Bach et al., 2016; Rekimoto et al., 2001) consist of relatively long training phases, rely on multiple pointer interactions with frequent repetitions of selection and parameter adjustments, lack simultaneous control of 2 parameters, present complex interfaces, are specifically designed for interacting with multivariate data or to navigate numerical tables, and used mostly for visualization purposes.



(A) (B)

Fig. 2. Examples of tile-based interfaces found in commercial software packages: (A) multi-dimensional domains (MD Slider®), and (B) material properties (Material Picker®).

To address all of the issues and limitations stated above, we developed the Rhomb-i slider: a set of rhombus shaped sliders that display several solutions of 3-D parameter domains and 2-D handles for real-time browsing and selecting. The proposed interface is directed towards interactive visualization of 3-D parameter domains, and is not specifically targeted for space-time visualization nor multivariate data. Note that, the proposed interface can be considered as a specialized sub-set of the Generalized Space-Time Cube (Bach et al., 2016), although it is more focused on browsing 3-D parameter domains than on space-time visualizations. Several differences distinguish both interfaces: Rhomb-i does not use the cube as a full solid object but only 3 adjoining facets that meet at a vertex; according to the taxonomy of elementary operations (Figure 24, (Bach et al., 2016)), explicit operations such as extraction, flattening, geometry transformation or content transformation are not performed via Rhomb-i interface, with the exception of point extraction and orthogonal cutting; point extraction is performed by manipulating one 2-D handle which will immediately affect the positions of the other two handles as they all share at least one common parameter.

Therefore, a generic tile-based interface is welcomed to reduce the dimensionality gap while improving navigation of 3-D parameter domains, even in constrained parameter domains or those that do not have an obvious spatial representation. Non-parallel configurations such as rhombille tiling patterns may provide better usability results for display, navigation and selection tasks since they are seemingly more natural to interact by using the widget surface area as a sketch-based interface. In particular, our proposed approach should be more efficient than 1-D sliders and present similar to or higher performance than specialized widgets. In addition, the clear affordances of the configurations should make it easy to learn and interact with.

3.1. From 1-D sliders to tile-based interfaces

One-dimensional sliders provide an abstract linear representation of a single quantity, and thus can handle a single parameter at a time. That parameter value can represent a discrete or continuous range of values, such as real-valued intervals $s_x \in [s_{x0}, s_{x1}] \in \mathbb{R}$, where the smallest, s_{x0}, and largest, s_{x1}, values appear on the left and right side, respectively (Fig. 3(A)). In general, continuous sliders are more common and should be considered whenever users do not need to set specific values which may require meaningful adjustments. In contrast, discrete sliders should be used for predetermined and properly spaced values placed along the slider rail, allowing a person to objectively select specific values.

Sliders provide simple interactions to set a value by direct manipulation, namely, by moving a handle or by clicking on a point on the slider rail. Usually, these sliders are complemented with non-visual information, namely the textual data referring to the numerical quantity associated to the slider's handle position, or visual information such as icons placed along the slider's length that display discrete values. As noted previously, this interface only provides a single visual datum per interaction (i.e., sliding a handle).

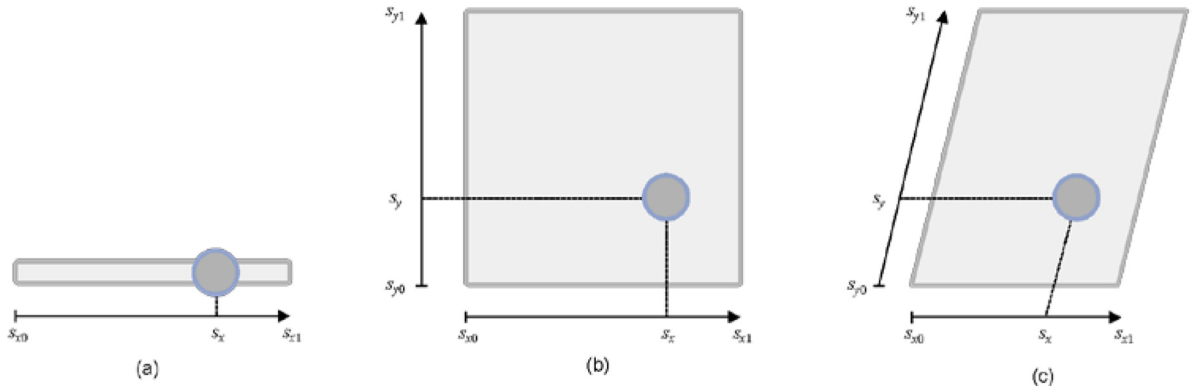


Fig. 3. One- and two-dimensional sliders: (a) linear slider, (b) rectangular tile slider, and (c) rhombus tile slider.

To navigate 3-D parameter domains on a flat medium, it is often necessary to use three sliders, one for each parameter, which typically present a linear grid arrangement (i.e., vertically stacked). More specifically, the interface must extend the one-dimensional slider into a two-dimensional surface patch or tile (Fig. 3(B) and (C)). A tile consists of a limited and closed set of points, with a quadrangular shape, on the planar surface and is constructed with two sliders. This offers both directions of pointer movement (e.g., mouse or touch) to control two values simultaneously.

To manipulate two parameters s_x and s_y, on an interval $[s_{x0}, s_{x1}]$x $[s_{y0}, s_{y1}] \in \mathbb{R}^2$, one can consider a couple of one-dimensional sliders, that can be accessed individually, or a quadrangular surface patch, also called a tile slider (Fig. 3(B), (C)). Since a tile slider is a planar entity, the handle can be positioned in 2-D parameter domain, giving simultaneous access to two parameters. This facilitates developing sketch-based interfaces where users can stroke a handle throughout the parameter domain. Tiles can be either rectangular or rhombus and only differ on the linear deformation: a rhombus is a skewed (or sheared) version of a rectangular tile along the x or y axis.

If a single handle in a tile gives access to two parameters, then by combining two or more tiles makes it possible to browse more coordinates in the parameter domain *simultaneously*. Then, a distorted rectangular tile, such as a rhombus, allows a more flexible geometry than its quadrangular homologue and a wider range of tile patterns.

In what follows, we propose the *Rhomb-i slider* whose design is inspired by the rhombille tiling pattern, i.e., a tessellation of the Euclidean plane composed by 60° rhombi (or diamonds) forming hexagons divided into three rhombi meeting at their center points (Fig. 4). The proposed visual interface projects 3-D parametric solutions onto a set of 2-D rhombi, where each edge corresponds to a single parameter or 1-D slider.

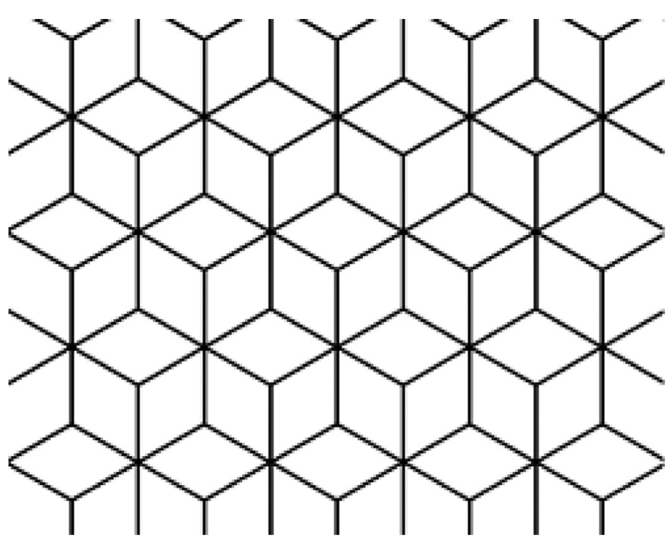


Fig. 4. Rhombille tiling pattern.

Note that this set of 3 contiguous rhombi forms a tiling pattern that can be interpreted as an isometric view of a cube, creating an ambiguous line drawing known as the Necker Cube or reversible cubes illusion (Necker, 1832). Thus, a Rhomb-i can also be interpreted as a configuration of 1-D sliders with a tripod arrangement formed at the common point where the three rhombi meet (Fig. 5).

In other words, if we consider each edge of a rhombus as a 1-D slider, then each rhombus can be interpreted as a tile slider. Equivalently, the connecting grooves form a tripod shape, which can be considered as an alternative slider arrangement or a deconstruction of the linear grid arrangement found in 1-D sliders (Fig. 5). Contrary to 1-D sliders, a Rhomb-i gives the user simultaneous access to manipulating up to two parameters of the parameter domain. When grouping 3 rhombi to form a rhombille tiling pattern, the user has then access to all 3 domain parameters through a bi-dimensional interface as shown in (Fig. 5(C)).

4. Comparing visual interfaces to navigate 3-D parameter domains

To assess the usability and to verify if the Rhomb-i slider is a generic design model to efficiently browse 3-D parameter domains, we compare the proposed tile-based interface with conventional 1-D sliders and specialized widgets. In this study, the measured variables are task completion time and participant preferences as quantitative and qualitative measures, respectively. Without loss of generality and in order to test the proposed tile-based interface, we explore three different parameter domain: HSV color space, geometric parameters of super-shape curves, and rotation angles of a 3-D object. Therefore, nine graphical user interfaces were implemented for the user study: (i) 1-D sliders for HSV color, (ii) 1-D sliders for super-shape, (iii) 1-D sliders for object rotation, (iv) specialized widget for HSV color, (v) specialized widget for super-shape, (vi) specialized widget for object rotation, (vii) Rhomb-i slider for HSV color, (viii) Rhomb-i slider for super-shape, and (xix) Rhomb-i slider for object rotation. All the code was developed in C# using the Unity game engine (version 5.32f1) to build real-time interaction applications. In this Section, the various parameters of the different visual interfaces are explained in detail, as this information is

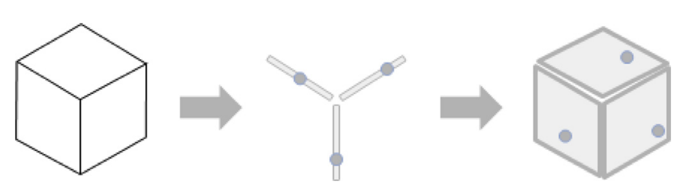


Fig. 5. Visual comparison between Rhombille tile pattern, 1-D sliders arranged in a tripod configuration, and Rhomb-i interface.

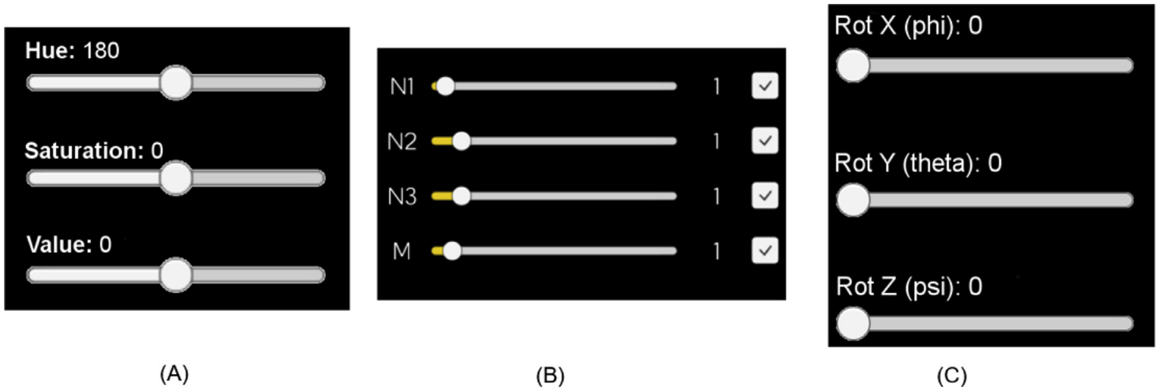


Fig. 6. 1-D sliders: (A) HSV color, (B) super-shapes, and (C) 3-D object rotation.

relevant for the usability assessments reported in the user study (Section 5).

4.1. 1-D sliders

The implementation of 1-D sliders relied on the UI Slider of the Canvas component available with the Unity game engine (version 5.32f1). The 1-D sliders were stacked vertically as this is a common arrangement found in many software packages (Fig. 6). The 3-D mesh object's color property, the exponents of the 2-D super-space curve and the 3-D mesh object's orientation were mapped to these sliders (Table 1). An extra 1-D slider for the *M* parameter of the super shape equation was added in all different interfaces, as it was a given parameter at the start of the task (Fig. 6(B)). (Table 1) indicates the interval of values for each parameter or, equivalently, for each 1-D slider.

4.2. Specialized widgets

As can readily be seen, each 3-D parameter domain has its own characteristic visual state depictions (Fig. 7). To this end, specialized widgets can be designed to explore specific visual metaphors that draw the most of the 3-D parameter domain nature, as they display a greater variety of design solutions when compared to the visually concise 1-D slider. (Fig. 7) illustrates specialized widgets for color picking and 3-D rotation. The specialized widgets considered in this study were chosen based on their popularity, i.e., they are commonly found in vector drawing, image editing, 3-D modeling, and image editing software, and due to their ease of computational implementation.

4.2.1. HSV color

The Hue–Saturation–Value (HSV) model (Smith, 1978) is known to be more intuitive and convenient to artists than the Red-Green–Blue (RGB) model (Foley, 1995). It is widely used in color selection widgets within vector drawing, 3-D modeling and image editing software. The hue parameter refers to the pure color within a color space and is described by a number on the color wheel ranging from 0° (red) and 360° (violet). In other words, it represents the color found in a rainbow. As for saturation, it quantifies the amount of purity contained within a color and is represented as a fraction between 0 (dull or desaturated) and 1 (bright or saturated). Value, also known as lightness, indicates how dark a color is where 0 corresponds to black and 1 to white.

A commonly found HSV color widget is composed by a 2-D slider that co-adjusts Saturation and Value, given the Hue fixed on a 1-D slider (Fig. 7(A)). Thus, once the hue value is altered, the visual depictions rendered on the 2-D slider are updated to carry the new hue value.

Within Unity game engine, this specialized widget was implemented by altering the 1-D slider asset background property to represent hue, and a 2-D texture was used to represent the value and saturation, where the color value was multiplied by the selected hue.

4.2.2. Super-shapes

Super-shapes or super-polygons are a versatile generalization of the superellipse that can represent idealized geometric primitives such as circles, ellipses, triangles, rectangles, N-gons, and a myriad of concave or convex shapes (Gielis, 2003). To wit, super-shapes consist of generalized polygons with the peculiar feature of presenting linear or nonlinear edges (Fig. 8). Besides the number of sides, the shape is controlled by three real-valued exponents that confer it either a convex or concave appearance. Its analytical expression can represent a myriad natural and manmade objects.

In polar coordinates, the parametric angle-center expression of the super-shape is given by

$$r(\varphi_s) = \left(\left| \frac{\cos\left(\frac{m}{4}\varphi_s\right)}{a} \right|^{n_2} + \left| \frac{\sin\left(\frac{m}{4}\varphi_s\right)}{b} \right|^{n_3} \right)^{-\frac{1}{n_1}}, \; 0 \le \varphi_s < 2\pi \tag{1}$$

where $r \in \mathbb{R}^+\backslash\{0\}$ and $\varphi_s \in [0, 2\pi[$ *are the polar coordinates of radius and angle, respectively; a,* $b \in \mathbb{R}^+\backslash\{0\}$ *are the radii dimensions along the* $x \in \mathbb{R}$ *and* $y \in \mathbb{R}$ *directions of the local Cartesian coordinates; m is the number of sides or edges;* $n_1, n_2, n_3 \in]0, +\infty[$ are the exponents that affect the convexity or concavity of the edge profiles.

Although no specific widget has yet been developed to explore super-shapes, such a 3-D parameter domain may benefit from having a slider with preview figurines, thumbnails or glyphs (Fig. 7(C)). This technique is useful to convey information as it encodes data by depicting markers placed at discrete locations in the parameter domain while helping users to build understanding. These depictions also reveal

Table 1
Continuous real-valued intervals of each parameter considered in this study.

	HSV color		Super-shapes		Rotation	
	Parameter	Interval	Parameter	Interval	Parameter	Interval
Top slider	Hue	[0.0, 360.0]	Exponent n_1	[1.0, 10.0]	x-axis	[0.0, 360.0]
Middle slider	Saturation	[0.0, 1.0]	Exponent n_2	[1.0, 10.0]	y-axis	[0.0, 360.0]
Bottom slider	Value	[0.0, 1.0]	Exponent n_3	[1.0, 10.0]	z-axis	[0.0, 360.0]

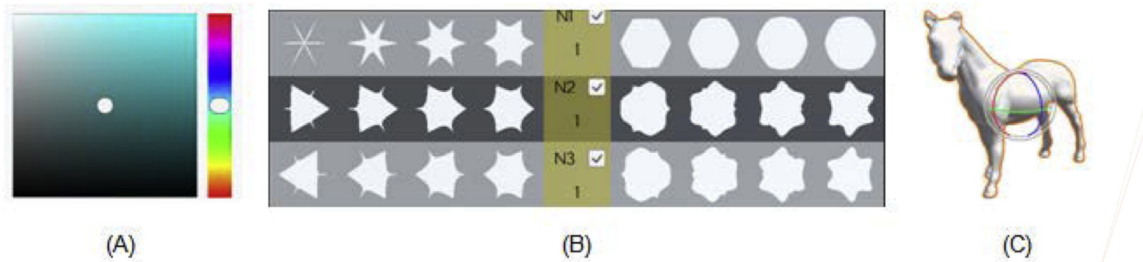


Fig. 7. Specialized widgets to explore emblematic 3-D parameter domains: (A) HSV color, (B) super-shape curves, and (C) object rotation.

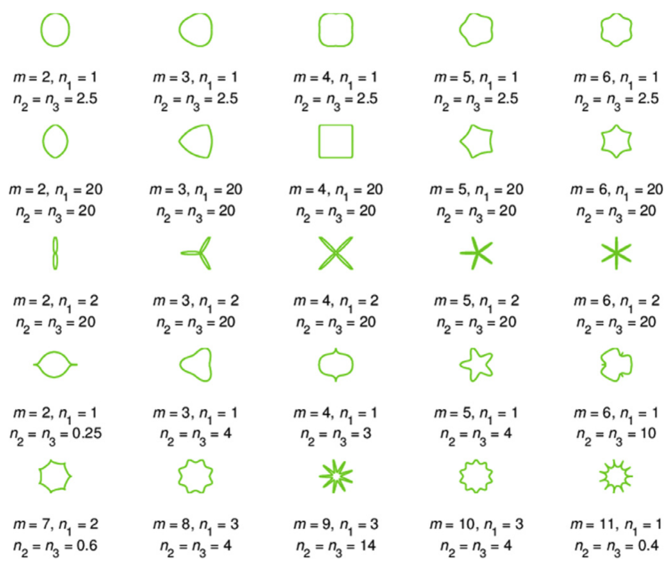


Fig. 8. Examples of super-shapes with varying values of number of sides and exponents.

patterns contained within the parameter domain (Schultz and Kindlmann, 2010; Ropinski et al., 2011; Li et al., 2015).

The super-shape figurines were implemented in Unity using (Eq. (1)) to sample each point of the different curve shapes. These curves follow a horizontal arrangement, hence, each slider consists of an array of miniature super-shapes that are placed upon the slider bar assets.

4.2.3. Rotation

In many different manipulation and visualization tasks ranging from CAD Packages to Virtual and Augmented Reality, when picking up a small object in one hand, the user can change its orientation in all three directions. The same interaction is ill-posed for a flat medium interface as it can only provide access, at most, to two independent directions at a time.

Rotating a 3-D object in a controlled manner, often implies a circular movement around an imaginary line denominated axis of rotation, for a given angular value. It consists of a rigid body motion which keeps, at least, a fixed point or, at most, a fixed line. Here, only axes passing through the object and around the principal rotations are considered. In particular, we consider the Euler angles which define a rigid body revolving around an arbitrary axis as a combination of principal circular displacements. These can be composed for instance, by a rotation around the x axis, followed by a rotation around the y axis and, finally around the z axis, where the corresponding angles of rotation of the x, y, and z axes are φ, θ, and ψ, respectively.

The most typical rotation widget found in 3-D modeling software packages consists of a set of orthogonal rings (Foley, 1995), one per canonical axis, that are aligned with local or global referential systems (Fig. 7(B)). Users can rotate along individual axis by picking the corresponding ring and reposition the handle to another point upon the ring. The amount of travelled perimeter defines the angle of rotation.

For the rotation parameter domain, we simply made use of Unity's default rotation widget that is characterised by three interactive orthogonal arcs, one for each rotation axis.

4.3. Rhombille tile-based interface and interaction

Rhombille tile-based interfaces can be tailored as interaction components for navigating different parameter domains (Fig. 9). This can be accomplished by overlaying visual aids on top of the tile or by recoloring the tile area. For each handle position on a given tile, the visual aids on the remaining tiles are redisplayed when the selected handle moves. Thus, during redisplay due to dynamic sliding, while the rendering is transient it still reveals the interdependencies between all adjacent sliders. Note that the tile area visually depicts the precision associated with each handle.

The Rhomb-i slider runs either on mouse-based or touch-based interfaces since it is designed for single-point interaction. The user can perform a single click/tap to select a unique solution or click -and-drag/tap-and-drag to browse throughout the parameter domain. Despite having access to three cursors, the user can only manipulate two parameters at once. Users may try to access a single parameter but usually encounter minute mouse imprecisions or fat finger issues. Thus, users activate a single cursor at a time (i.e., one of the yellowish hemispheres in Fig. 9). Bi-manual interaction is not possible because all three cursors are interlinked in the following manner: by changing the coordinates of one cursor will immediately affect the positions of the other two cursors as all cursors share at least one parameter.

The Rhomb-i sliders were implemented in Unity using three different planes positioned orthogonally to each other, and a relatively small sphere (yellowish handle in (Fig. 9) that had its coordinates constrained to the surface of one of the planes to work as a cursor. For the HSV space, a HLSL shader was implemented to display the color distribution at each surface. For super shapes, miniatures of the resulting shapes for each point of the 2-D rhombus slider were rendered in a 3 × 3 regular grid. Finally, the rotation Rhomb-i represented a set of miniature 3-D objects covering the 2-D rhombus sliders and placed in 3 × 3 regular grids, each with a different orientation according to the underlying angle parameters. A camera with orthographic projection was placed facing the common vertice to give an isometric or rhombille tiling pattern appearance to the 3-D construction.

5. User study

A user study was conducted to assess the usability of the Rhomb-i slider for visual navigation of three distinct parameter domains: HSV color, super-shape curves, and rotation of 3-D objects. In addition to the Rhomb-i sliders (Fig. 9), we also implemented graphical user interfaces using 1-D sliders (Fig. 6) and specialized widgets (Fig. 7).

For the purpose of this user study we declare independent variables based on the parameter domain and visual interface, which gives a total of nine independent variables or, by other terms, nine graphical user interfaces: (i) 1-D sliders for HSV color, (ii) 1-D sliders for super-shape,

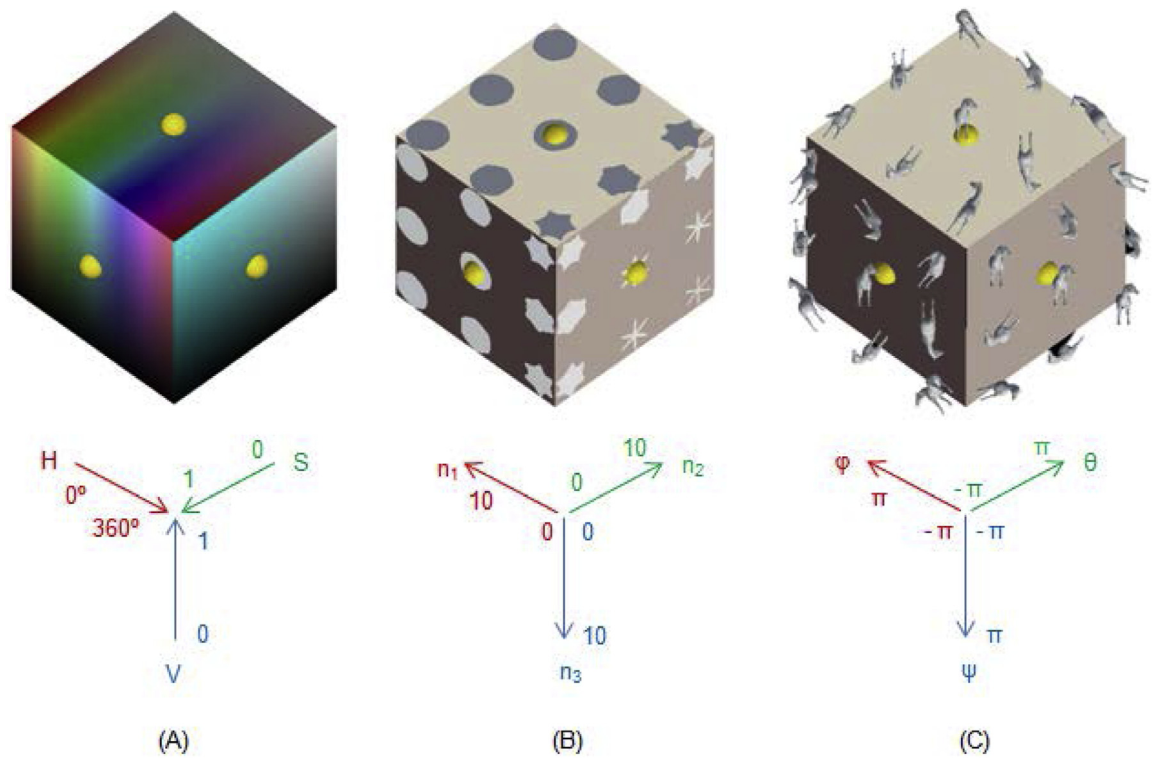


Fig. 9. Rhomb-i configurations and parameter mappings for (A) HSV color space, (B) super-shape and (C) rotation 3-D parameter domains. Note that each tile has a handle (orange spherical cap) that users pick and slide along the planar parametric domain, while the positions of the other two handles of the adjacent tiles and associated tile attributes (e.g., color space attributes or figurines) are updated. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

(iii) 1-D sliders for object rotation, (iv) specialized widget for HSV color, (v) specialized widget for super-shape, (vi) specialized widget for object rotation, (vii) Rhomb-i slider for HSV color, (viii) Rhomb-i slider for super-shape, and (xix) Rhomb-i slider for object rotation. In order to cope with the goals of our study, we declare task completion time and participant preferences as dependent variables, which are quantitative and qualitative measures, respectively. Such dependent variables will allow to verify if the proposed tile-based interface is more efficient and less time-consuming for users to search the 3-D parameter domain on the proposed tasks, and if it is a preferable slider compared to other conventional interfaces.

Our goal was to verify if the proposed tile-based interface is more efficient and less time-consuming for users to search the 3-D parameter domain on the proposed tasks, and if it is a preferable slider compared to other conventional interfaces. Task completion time and participant preferences are measured for this purpose.

Note that the design of the study is organized according to the considered dependent and independent variables: for each graphical user interface (i.e., total of 9), participants were asked to perform several tasks (i.e., 3 per graphical interface, which gives a total of 27 tasks per participant). All tasks are timed and participant preferences are also measured. Finally, code was developed in C# using the Unity game engine (version 5.32f1) to build real-time interaction applications.

5.1. Participants

A total of 19 unpaid invited participants performed the user study, with ages between 20 and 49 years old (63% between 20–24 years old, 15 male and 4 female). None of the participants reported to be color blind. Regarding academic background, 14 had at least a bachelor's degree in Computer Science, 2 had a bachelor's degree in Social Sciences, and 3 participants had a doctoral degree also in Computer Science. Participants were selected to match the user profile for parameter domain navigation, namely, they were knowledgeable regarding 3-D parameter domains. For instance, all participants have used visual interfaces for exploring 3-D parameter domains of HSV color and rotation, although none of the participants ever crossed with super-shapes before. In particular, all of the participants were familiar with the traditional specialized widgets but none of them were familiar with the super-shape parameter domain nor with the Rhomb-i sliders used in the experiment. Image editing software is used daily by 31% of the users, and 21% operates with 3-D modeling software in a daily basis.

5.2. Apparatus

The study was taken place in a quiet office, on a desktop computer (i7 3.40Ghz 16GB ram, Nvidia GeForce GTX 980 Ti) equipped with keyboard, mouse and a 17 inch display with a resolution of 1920×1080.

5.3. Tasks

The test sessions were individual and each task was timed. Task completion time is the considered performance measure. The expected duration of a test session was about 30–40 minutes and was divided in three phases (one per parameter domain), where each phase required the completion of three tasks per visual interface. Thereby, all participants were asked to interact with the nine graphical user interfaces. With each interface, participants completed three different tasks.

Before each navigation task (Fig. 10), users were asked to perform an habituation task which consisted of interacting with the displayed interfaces (Fig. 11) by browsing through the domain values so that the user could get used to the interface and its mechanics. A total of 9 habituation tasks (3 parameter domains x 3 user interfaces per parameter domain) were performed.

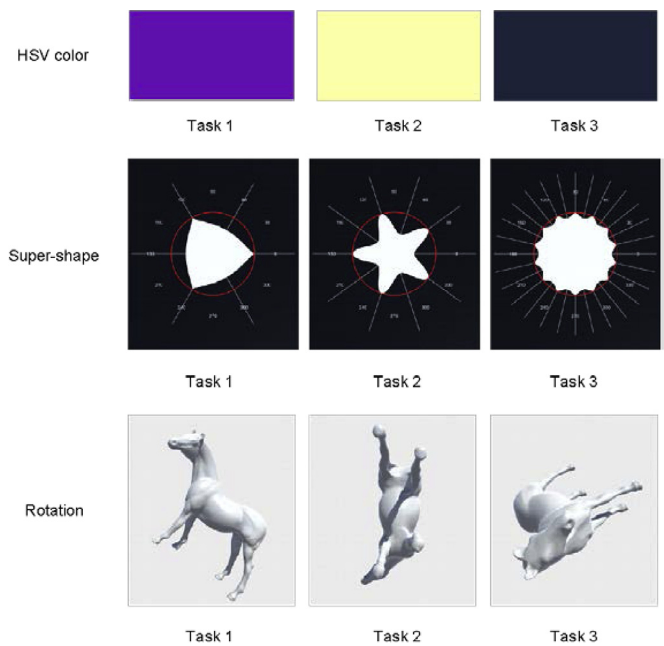


Fig. 10. User tasks for exploring 3-D parameter domains. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

During each navigation task, either HSV color, rotation, or super-shape domains spaces were tested. As the control interface, we considered a set of three 1-D sliders vertically arranged since this is a conventional interface for 3-D exploration found in many software packages. We asked participants to complete several navigation tasks using different visual interfaces. Sets of three colors, rotation configurations and super-shape figures were presented to the users who were requested to find the desired visual solution (Fig. 10) using the graphical user interfaces with 1-D sliders, specialized widgets and Rhomb-i widgets (Fig. 11), which totalized 27 navigation tasks (3 parameter domains x 3 user interfaces per parameter domain x 3 desired visual solutions).

The interfaces used for the tests included only the controls for the designed task, and a start and finish button to control the test (Fig. 11). Task completion time was measured since the beginning of the task until the user was satisfied with the result. Each task could take at most 2 min, after which the task was interrupted.

5.4. Procedure

At the beginning of the session, participants filled in an informed consent form agreeing to the study conditions and terms. Before starting the tasks, it was necessary to draw which 3-D parameter domain and which visual interface a user would test first. To obtain a as balanced as possible number of participants per initial visual interface and initial task, we ordered the combinations by resorting on Latin squares' permutations. Thus, each participant tested a different combination of parameter domain, visual interface, and task. Note that this is a temptative form to prevent biased results or any effect of order of the tested parameters. Then, at the beginning of each phase, a short system presentation was made in order to explain how each objectified functionality worked. Afterwards, the user would test the visual interfaces up to 2 minutes per task after a period of habituation. At the end of each phase, each participant was asked to complete a questionnaire regarding the visual interfaces and about the tasks undertaken in order to classify the level of difficulty felt during their tasks performance and on the use of the available features (Tables 2 and 3). The full set of questions, which are related to usefulness, easiness and recallability are listed in Tables 2 and 3. The time required for task completion time was also measured (Fig. 12).

6. Results

This section begins by reporting user study measurements in two separate sub-sections according to the quantitative and qualitative nature of the dependent variables under consideration, namely, task completion times and participant preferences, respectively.

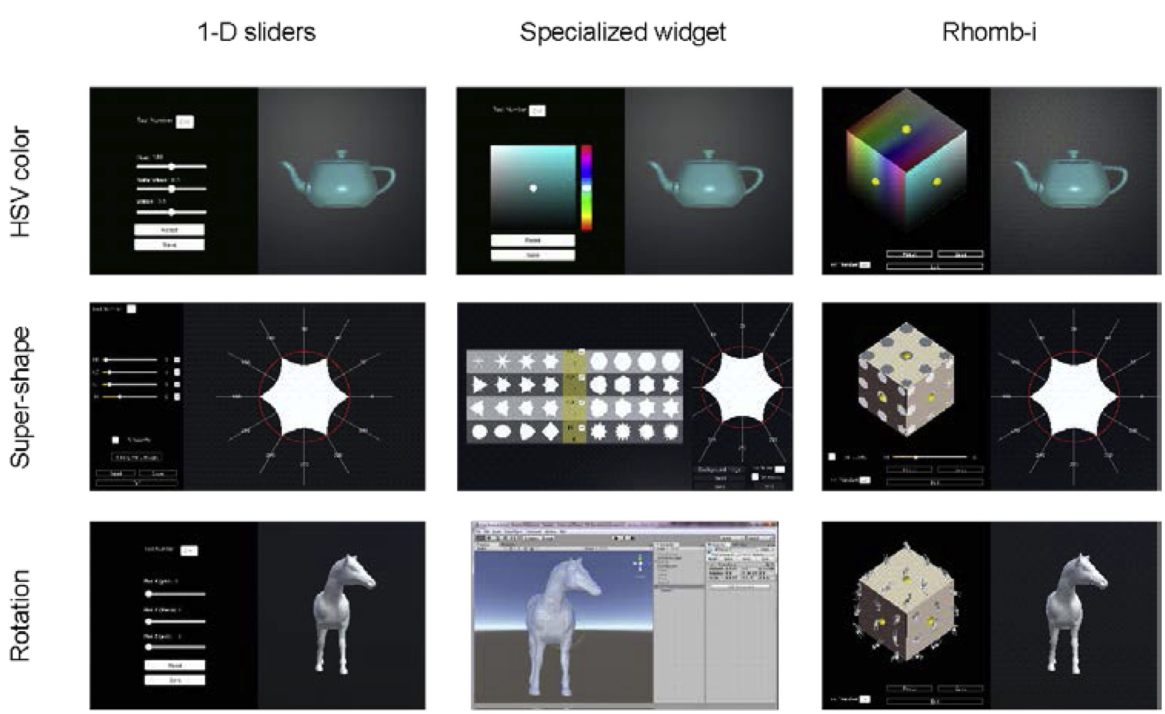


Fig. 11. The graphical user interfaces considered for the user study. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 2
Participants preferences regarding different criteria for the 1-D Sliders (1-D), Specialized Widget (SW), and Rhomb-i (R-i) sliders: Median (interquartile range). Likert scale: 1 – totally disagree and 6 – totally agree.

	HSV color			Super-shapes			Rotation		
	1-D	SW	R-i	1-D	SW	R-i	1-D	SW	R-i
In general, it was easy to use the interface.	4(2)	5(2)	5(1)	4(1.5)	4(2)	4(2)	5(1)	5(0.75)	2(1)
In general, it was easy to explore the parameter domain.	4(2)	5(0)	5(1)	4(1.5)	4(2)	4(2)	4(2.75)	5(2)	3(3.75)
It was easy to find the requested color/orientation/shape.	5(1)	5(1)	5(1)	5(1.5)	5(0.5)	5(2)	6(1)	6(1)	4(2)
The spatial configuration of sliders was appropriate to its function.	5(2)	5.5(1)	5(1)	5(1.5)	4(2)	5(1)	5(1)	6(1)	4(2)
It was easy to identify how the control buttons worked.	6(1)	6(1)	6(1)	5(1.5)	5(1)	5(1.5)	6(1)	5(1.75)	4(2.5)
The interface has a coherent layout.	6(1)	6(1)	6(1)	5(1)	5(2.5)	4(2)	5.5(1)	5(1)	5(1.75)
The interface is comfortable to use.	5(2)	6(0.5)	5(2)	4(1)	4(2.5)	4(2.5)	5(1)	5(1.75)	3.5(4)
The interface was easy to learn.	5(1.5)	6(1)	5(1.5)	4(2)	4(2)	3(2.5)	5(1.5)	5(1.75)	2.5(3)
It was quick to complete the requested task.	5(1)	4(2)	6(1.5)	3(2.5)	4(2)	4(2)	4(1)	5(1)	2(1)

Measurements were performed for all nine independent variables that are defined by the visual navigation variable (i.e., HSV color, geometric parameters of super-shape curves, and rotation angles of a 3-D object) with respect to each visual interface type (i.e., 1-D sliders, specialized widget, Rhomb-i). This section ends with observations on interaction behaviour relative to the visual interfaces.

6.1. Task completion times

Task completion times for each independent variable can be seen in (Fig. 12). Statistical significance was tested using the Friedman test ($p < 0.05$), and corroborated with the task completion time results by a pairwise Wilcoxon test, using Holm-Bonferroni correction. Task completion times were found not to have a normal distribution according to a Shapiro-Wilk test. Despite the similarity of task completion times for all of the proposed HSV color interfaces, statistical significances were found between 1-D sliders and Rhomb-i slider ($p = 0.039$), 1-D sliders and specialized widget ($p = 0.02$) and specialized widget and Rhomb-i slider ($p = 0.002$). Regarding super-shape domain, task completion time results were statistically significant when comparing 1-D sliders and Rhomb-i ($p = 0.002$) and specialized widget with Rhomb-i ($p = 0.001$). For the Rotation tasks, no difference was found using the pairwise Wilcoxon test when comparing 1-D sliders to the specialized widget or the Rhomb-i slider, but was found when comparing specialized widget to the Rhomb-i slider ($p = 0.001$).

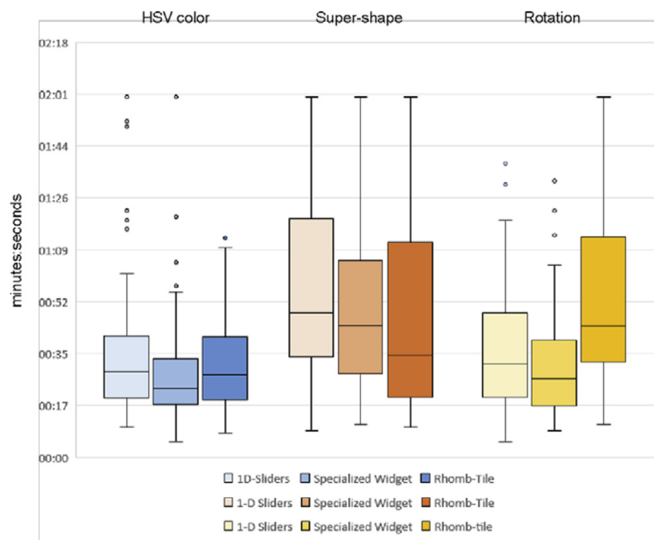


Fig. 12. Task completion times for the considered 3-D parameter domains. Markings of the box plots correspond to minimum, interquartile range, mean, and maximum. Individual points correspond to outliers.

6.2. Participant preferences

Regarding participant preferences (perceived satisfaction), where we used a Likert scale (1 – 6), these can be found in Tables 2–4. Participant preferences were also found not to have a normal distribution according to a Shapiro–Wilk test. When compared to the user questionnaires applied at the end of each task, we found that the user perception of the time spent on each task (Table 2) was aligned with the actual recorded values (Fig. 12). Either tile-based interfaces or

Table 4
Overall participant preference regarding the 1-D Slider (1-D) and Specialized widget (SW) interfaces: Median (interquartile range). Likert scale: 1 – totally disagree and 6 – totally agree.

	HSV color		Super-shapes		Rotation	
	1-D	SW	1-D	SW	1-D	SW
Although unusual, I consider the Rhomb-i slider to be a viable alternative to …	5(2)	4(2)	6(3)	5(3)	2(2.75)	1(2)

Table 3
Pairwise significant p-values of participants preferences regarding different criteria for the 1-D Sliders (1-D), Specialized Widget (SW), and Rhomb-i (R-i) sliders.

	HSV color			Super-shapes			Rotation		
	1-D|SW	1-D|R-i	SW|R-i	1-D|SW	1-D|R-i	SW|R-i	1-D|SW	1-D|R-i	SW|R-i
In general, it was easy to use the interface.	–	–	–	–	–	–	–	0.001	0.0001
In general, it was easy to explore the parameter domain.	0.014	–	–	–	–	–	–	–	–
It was easy to find the requested color/orientation/shape.	–	–	–	–	–	–	–	0.001	0.001
The spatial configuration of sliders was appropriate to its function.	–	–	–	–	–	–	–	0.003	0.003
It was easy to identify how the control buttons worked.	–	–	–	–	–	–	–	0.011	–
The interface has a coherent layout.	–	–	–	–	–	–	–	0.011	–
The interface is comfortable to use.	–	0.003	–	–	–	–	–	0.002	0.003
The interface was easy to learn.	–	0.004	–	–	–	–	–	0.001	0.001
It was quick to complete the requested task.	–	–	–	–	–	–	–	0.001	0.001

specialized widgets provided improved performance (i.e., lesser navigation time) regarding 1-D sliders. The fastest task performing interfaces received higher overall ratings also on perceived quality of the achieved results.

6.3. Visual interface interaction

All visual interfaces enabled the selection of similar solutions. The results also indicate that the proposed tile-based interface delivers different probing experiences for the parameter domains under consideration. Yet, participants rapidly understood how to interact through the visual interfaces, in particular the Rhomb-i which never has been used before. In addition, all participants considered Rhomb-i sliders may be useful in graphic design, game design, image editing and 3-D modeling software packages.

7. Discussion

Rhomb-i slider is our novel rhombille shaped tile-based interface to improve real-time browsing and selection within 3-D parameter domains. . We assessed its usability to visually navigate HSV color, super-shape curves and 3-D object rotation parameter domains by verifying navigation efficiency and its ability to represent parameter domains of varying complexity.

The motivation behind Rhomb-i slider design (i.e., the re-configuration 1–D sliders into a rhombille pattern) consists of adding a third dimension to allow more information to be simultaneously displayed on the screen. Due to its geometric arrangement (i.e., three 1-D sliders aligned along the edges of a rhombus pattern) it is possible to interact with 2 parameters simultaneously, with the added capacity of transitioning from one tile to the next by simply crossing an edge/slider. It is even possible to enhance patterns contained with the parameter domain space (e.g., rainbow in the HSV space (Fig. 9(A))).

To assess whether Rhomb-i is a generic design model to efficiently browse 3-D parameter domains, we considered graphical objects with varying degrees of visual complexity and inquired what characteristics make Rhomb-i sliders better suited and, beyond which, require more sophisticated or specialized design interfaces. Users dealt with more and more complex visual objects (a single color, a 2-D planar shape, and a 3-D object). In what follows we discuss the usability and potential limitations of the Rhomb-i slider for each parameter domain from simplest to the more visually complex.,

HSV color: although three participants were not familiar with HSV color and the meaning of each one of the individual values, all participants reported to use image manipulation software at least once a month, with several being frequent users of a typical color picker. Color selection via 1-D sliders required more task completion time due to the fact that participants usually do not pick colors using sliders, since this demands proper color weighting and balancing. Because color appearance depends on comparison of neighboring colors, the Rhomb-i interface clarified the relationships between colors. Overall, all visual interfaces offered very similar experiences to participants, which show that Rhomb-i's approach, albeit less familiar, was at least at par with the very common HSV specialized widget.

Super-shape: being an unfamiliar parameter domain for all participants and one with non-linear relationships between parameter values and expected design solution (Gielis, 2003), the interface with shortest task completion times can be considered as the most appropriate visual encoding for super-shape curves. In general, user questionnaires showed statistical significance in most of the Rotation and Color related questions, but did not show with super-shapes. This was likely due to the fact that when performing a task in a new parameter domain, user opinion was not yet fully formed. The Rhomb-i interface allowed to easily modify the exponent parameters and for the perceptual monitoring of a large number of visual solutions. Since each tile is an interactive surface, it enabled the exploration of a 2-D subset of the parameters. Tiles represent a large amount of visual solutions in a small domain, which much lead to a search time reduction. Results clearly favored the Rhomb-i interface which also had the highest ranking on all questions on the user questionnaire, which confirmed our hypothesis of Rhomb-i being well suited for unfamiliar parameter domains. Due to its non-linear expression, super-shape curve space can be very complicated to visually navigate, since there is no direct mapping between the numerical value and expected shape for certain parameter intervals (Barrios, 2014). The Rhomb-i interface provides several visual solutions per interaction (i.e., sliding a handle), hence, it takes proper advantage of the human eye's bandwidth for exploring and understanding multiple design solution at once.

Rotation: rotation performance varied widely between subjects. Participants performed better with the specialized widget either because it was placed on the object itself or because participants were already familiar with the interface. Interestingly, participants performed considerably worse on the Rhomb-i interface. During the tests, we found that participants would more naturally perform rotations by applying sequential single axis transformations, a situation better achieved with 1-D sliders and by dragging the arcs of the specialized widget. Moreover, as reported by Johansson et al. (2014), visualizing 3-D patterns in a 2-D medium does hamper users' judgment on recognizing patterns in 3-D. This may have been the case as the orientation of an object is a 3-D pattern. It may be necessary to redesign the graphical user interface to present a more natural mapping of coordinates and transformation directions, e.g., 1-D sliders arranged in a tripod configuration showing their handles (see Fig. 5(B)) can be added to explicitly indicate that 1-D sliders are in fact available at this interface. Also, this interface had a small granularity of samples on each tile to guide people while exploring the three-dimensional domain, forcing participants to advance by trial-and-error. Few participants were able to use the Rhomb-i interface at its full potential as they realized that single axis transformations could be performed independently and combined to achieve the desired rotation. Therefore, results indicate that the Rhomb-i design requires further improvements as most people took too long to achieve the desired rotation. Improved outcomes could be obtained by considering more natural axis arrangements or by rendering more figurines or glyphs per tile.

User feedback revealed that Rhomb-i, albeit less familiar, presents a good synthesis capacity and interpretative clarity. It was also sufficiently abstract to represent domains spaces with distinct visual complexities (e.g., HSV color and super-shape curves). Since Rhomb-i results from the ordered grouping of several tiles, such visual interface presented an aesthetic design which provided participants with a cohesive and common language for interaction, maintaining consistency with conventional 2-D sliders.

Despite the contributions of our work, the performed study still represents a basic first attempt to assess the effectiveness and efficiency of the proposed 3-D parameter domain selection interface for HSV color, super-shapes and 3-D object rotation, hence, the results obtained reveal that the current version of Rhomb-i can be considered as an initial concept validation for these parameter domains.

Towards generic design limitations, Rhomb-i provides a wider area to place visual solutions also carries intrinsic problems, namely, it showcases more content to visualize and analyze, hence, a greater cognitive load. Antagonistically, projecting 3-D parameter domain space into a flat medium brings occlusion of potentially useful information that is not instantly accessible since it is virtually impossible to render an entire real-valued 3-D domain space into a flat medium. Another crucial feature that may influence the relative usability of the Rhomb-i interface is the inherent tile distortion, i.e., data is represented on skewed quadrangular input surfaces producing a distorted view of the data. Additionally, as the solution to be found gets more complex, users may have more difficulty ascertaining a solution under a 2-D co-adjustment of parameters given the third parameter fixed, as this may impose heavier cognitive burden to users as visual objects get more complex. Such predicament suggests that the Rhomb-i slider may not be easily scalable to N-dimensional parameter domains ($N \geq 4$). Another limitation arises whenever parameters have non-linear correlations with the solutions, as is the case of super-shape curves. This may make it harder for users to orient where they are in the search space, since that is harder to tell from the linearly adjusted slider and non-linearly changed solutions. Therefore, further work must be performed to deal with more complex objects and parameter domains.

Such limitations are corroborated by user feedback, which point out several design issues that should be reconsidered. Regarding the design of the Rhomb-i slider for HSV color, one participant mentioned that choosing the hue value was very easy; yet finding the right saturation and value was not as simple as the lateral tiles were being updated in real-time. Another participant remarked that the value axis of the color Rhomb-i slider (Fig. 9(A)) could be larger to facilitate color picking. Two participants also indicated that the handles could be colored according to the current selection. As for the super-shape Rhomb-i slider, although participants appreciated the use of several figurines on all three tiles as it provided a better understanding of the parameter domain, three participants were slightly confused by the constants updates occurring on the non-active tiles ``*as the figurines changed in ways I was not expecting*” . Other participant specifically mentioned that it was ``*easier to operate with 1-D sliders because the separated parameter gave me a better sense of controllability*”. Several Rhomb-i redesign suggestions were also provided for the rotation of a 3-D object. Although several participants felt slightly intimidated at first, they admitted that after explaining how the Rhomb-i slider works it ``*was quite easy and interesting to handle*”. Interestingly, one participant indicated that the handles could be replaced by a figurine of the horse in the selected orientation.

Regarding user study methodology, the matching tasks under evaluation were cognitively demanding because of their numerical nature: participants were asked to navigate and select a predefined visual solution that was within a 3-D parameter domain that, in turn, is a set of real-valued continuous intervals with high numerical resolutions (Table 1). Hence, each parameter domain offers a virtually infinite solution space. Inherently, different participants have different stopping criteria. Since it was up to the participant to decide whether he/she reached the desired visual solution, the measured task completion times could be more accurate if (i) the task specified which parameter(s) needed adjustment (e.g., display of numerical values of each parameter below both the initial and final visual objects); (ii) the task notified the participant when it was within range of the target value (e.g., with a mark next to each parameter or by applying a color cue to the handle); and/or (iii) once all the parameters are within range, the task would immediately end. Such could avoid users to change a parameter that was already at the correct target value or to make the task much less complicated (Sriranga and Raju, 2013). However, even if we adopted these experimental configurations, users could be tempted to randomly browse the parameter domain until reaching the solution without properly testing the mechanics of the Rhomb-i slider. Despite this task completion time caveat, the performed usability study followed good interaction practices and the questionnaires reveal participants preferences that corroborate with the task completion times.

Even though participants tested the same visual interfaces several times, the user study did not evaluated the effect of learning. It should be noted that evaluating the learning process was not a part of our objectives since each user interacted with the independent variables in a different order, and we observed no difference in their performance directly related to any specific ordering.

A limitation of the user study was that it only tested with 19 subjects. A greater number participants would provide more statistical significance. It should be noted that we did not have access to an entire statistical population of interest, since it is difficult to find a large number of participants. Moreover, the evaluation process is too time-consuming to allow more than a small segment of the population to be observed. Nevertheless, we do recognize that a larger sample data set would provide more solid statistical evidence to support our conclusions. Even so, the results were validated from collected feedback and points towards new research paths on how Rhomb-i sliders can assist visual navigation of 3-D parameter domains.

8. Conclusions and future work

This work addresses navigation of 3-D parameter domains using visual interfaces as design tools in Computer Graphics applications. Based on the visual metaphor encountered in rhombille tiling patterns, were each edge of a rhombus can be interpreted as a 1-D slider, we present a novel tile-based interface called Rhomb-i. We assess the usability of this interface for probing and selection tasks so that users can more easily find appropriate design solutions within 3-D parameter domains. To this end, we conducted a usability study on visual navigation of 3-D parameter domains structured according to visual object complexity, namely, HSV color, super-shape curves, and rotation of a 3-D object. The proposed Rhomb-i slider is compared to 1-D sliders and specialized widgets. We verified that Rhomb-i sliders are a less time-consuming navigation approach for super-shape curves, offer similar navigation times for HSV color but are less efficient to find the orientation of a 3-D object. Regarding the ability of the Rhomb-i interface to unify specialized widgets for 3-D visual navigation, participant preferences clearly reveal that a single layout of three rhombus shaped 2-D sliders is able to represent different 3-D parameter domains, however, each Rhomb-i still requires highlighting salient features inside the 3-D parameter domain by resorting on visual aids. More specifically, our findings indicate that Rhomb-i is more appropriate for super-shape curves, has similar performance for HSV color but it did not cope as well for rotation of a 3-D object. In conclusion, the current version of Rhomb-i can be considered as an initial concept validation.

As future work, we aim to conduct a more in depth study to properly validate interaction precision and accuracy (i.e., measure the fiddling to obtain a desired selection or parameter state). To this end, a more elaborate user study with professionals (e.g., visual artists, graphic designers, architects) should be conducted. Besides task completion time and participant preferences, more complex measures to better account for user performance should also be considered, namely, heatmap of pointer trajectory, number of interactions or sliding events, and statistical analysis of the error between desired and selected solution.

Moreover, it is necessary to cope with Rhomb-i's current graphical design limitations. We also expect that the Rhomb-i slider in the form of color picker to be implemented in Computer Graphics or Computer-Aided Design software packages. Several interesting 3-D parameter domains found in Computer Graphics are worthy to be navigated via a Rhomb-i interface, such as scale, translation, RGB color, camera parameters (aperture, shutter speed, ISO), mechanical joint range of motion, contact parameters (elasticity, damping or friction), non-linear deformations, 3-link kinematic definition (e.g., arms and legs), character or facial animation constraints (e.g., eyes, arms, elbows and knees, hand, pelvis, feet, lips, eyebrows). In addition, other tiling patterns may serve as inspiration of developing alternative slider designs or arrangements. Since a single tile can be considered as a modular unit of interaction, if we consider several adjoining rhombille tiling patterns (i.e., a panel composed by four or more Rhomb-i) it is possible to explore N-dimensional parameter domains, enabling users to author simple graphical user interfaces to explore complex data sets through direct manipulation.

Acknowledgments

All authors are thankful for the financial support given by Portuguese Foundation for Science and Technology (FCT). In particular, the first author thanks for the postdoctoral grant SFRH/BPD/97449/2013 and the second author is grateful for the doctoral grant supported by the http://dx.doi.org/10.13039/501100000781 under the project (Ref. 336200) - ``BlackBox - A collaborative platform to document performance composition: from conceptual structures in the backstage to customizable visualizations in the front-end". This work was also partially supported by national funds through FCT with reference UID/CEC/50021/2013 and IT-MEDEX PTDC/EEI-SII/6038/2014. The authors would also like to thank David Dias for his work on the preliminary interfaces for super-shape exploration, and are grateful to Rui de Klerk and Eduardo Castro e Costa for their insightful comments about Rhomb-i Slider applications in Computer-Aided Design.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.ijhcs.2018.05.005.

References

Bellman, R.E., 2003. Dynamic Programming. Courier Dover Publications.

Fekete, J.D., 2004. The InfoVis toolkit. In: Proceedings of the IEEE Symposium on Information Visualization (INFOVIS '04). IEEE Computer Society, Washington, DC, USA, pp. 167–174. http://dx.doi.org/10.1109/INFOVIS.2004.64.

Reisman, J.L., Davidson, P.L., Han, J.Y., 2009. A screen-space formulation for 2D and 3D direct manipulation. In: Proceedings of the 22nd annual ACM symposium on User interface software and technology (UIST '09). USA. ACM, New York, NY, pp. 69–78. http://dx.doi.org/10.1145/1622176.1622190.

Yu, L., Svetachov, P., Isenberg, P., Everts, M., Isenberg, T., 2010. FI3D: DirectTouch, interaction for the exploration of 3D scientific visualization spaces. IEEE Trans. Visual. Comput. Graph. Inst. Electr. Electron. Eng. 16 (6), 1613–1622.

Jankowski, J., Hachet, M., 2013. A survey of interaction techniques for interactive 3D environments. Eurographics 2013. STAR, Girona, Spain *May 2013*.

Isenberg, T., 2016. Interactive exploration of three-dimensional scientific visualizations on large display surfaces. Collaboration Meets Interactive Spaces. Springer To appear.

Perin, C., Dragicevic, P., 2014. Manipulating multiple sliders by crossing. In: Proceeding IHM '14 Proceedings of the 26th Conference on l'Interaction Homme-Machine, pp. 48–54 Pages.

Damaraju, S., Seo, J.H., Hammond, T., Kerne, A., 2013. Multi-tap sliders: advancing touch, interaction for parameter adjustment. In: Proceedings of the 2013 International Conference on Intelligent User Interfaces. New York, NY, USA. pp. 445–452. http://dx.doi.org/10.1145/2449396.2449453.

Sriranga, D., Raju, S., 2013. An Exploration of Multi-touch Interaction Techniques. Texas A & M University.

Elmqvist, N., Stasko, J., Tsigas, P., 2008. DataMeadow: a visual canvas for analysis of large-scale multivariate data. Inf. Visual. 7 (1), 18–33. March. http://dx.doi.org/10.1145/1391107.1391110.

Satou, T., Kojima, H., Tonomura, Y., Akutsu, A., 2003. Scheme for Graphical User Interface using Polygonal-Shaped Slider. Google Patents April 1st, US Patent 6,542,171.

Schneider, S.E., 2010. Notched Slider Control for a Graphical User Interface. Google Patents January 7th, US Patent App. 12/168,644.

Tsandilas, T., Bezerianos, A., Jacob, T., 2015. SketchSliders: sketching widgets for visual exploration on wall displays. In: Proceeding CHI '15 Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems. Seoul, Korea. pp. 3255–3264 Pages.

Eick, S.G., 1994. Data visualization sliders. In: Szekely, Pedro (Ed.), Proceedings of the 7th annual ACM symposium on User interface software and technology November 02 - 04, 1994. Marina del Rey, California, United States, pp. 119–120.

Barrios, C.R., 2014. Parametric visualization and navigation in multidimensional spaces - a multidimensional matrix structure for navigation and visualization of parametric modeling instances, rethinking comprehensive design: speculative counterculture. In: Proceedings of the 19th International Conference on Computer-Aided Architectural Design Research in Asia. CAADRIA, Kyoto, pp. 543–552 14-16 May 2014.

Johansson, J., Forsell, C., Cooper, M., 2014. On the usability of three-dimensional display in parallel coordinates: Evaluating the efficiency of identifying two-dimensional relationships. Inf. Visual. 13 (January (1)), 29–41. http://dx.doi.org/10.1177/1473871613477091.

Bach, B., Dragicevic, P., Archambault, D., Hurter, C., Carpendale, S., 2016. A descriptive framework for temporal data visualizations based on generalized space-time cubes. Comput. Graph. Forum (to appear). http://onlinelibrary.wiley.com/doi/10.1111/cgf.12804/full.

Rekimoto, J., Ullmer, B., Oba, H., 2001. DataTiles: a modular platform for mixed physical and graphical interactions. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '01). ACM, New York, NY, USA, pp. 269–276. http://dx.doi.org/10.1145/365024.365115.

Tedeschi, A., 2014. AAD Algorithms-Aided Design. Parametric Strategies using Grasshopper. Le Penseur, Potenza, Italy.

Necker, L.A., 1832. Observations on some remarkable optical phaenomena seen in Switzerland; and on an optical phaenomenon which occurs on viewing a figure of a crystal or geometrical solid. London Edinburgh Philosoph. Mag. J. Sci. 1 (5), 329–337. http://dx.doi.org/10.1080/14786443208647909.

Smith, A.R., 1978. Color gamut transform pairs. Comput. Graph. 12 (3), 12–19. http://dx.doi.org/10.1145/965139.807361. August.

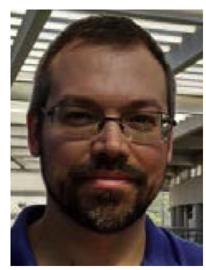
Foley, J.D., 1995. Computer Graphics: Principles and Practice, second ed. Addison-Wesley, Redwood City, CA ISBN 0-201-84840-6.

Gielis, J., 2003. A generic geometric transformation that unifies a wide range of natural and abstract shapes. Am. J. Botany 90 (3), 333–338. http://dx.doi.org/10.3732/ajb.90.3.333. ISSN 0002-9122.

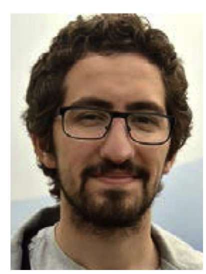
Schultz, T., Kindlmann, G.L., 2010. Superquadric glyphs for symmetric second-order tensors. IEEE Trans. Visual. Comput. Graph. 16 (November/December (6)), 1595–1604. http://dx.doi.org/10.1109/TVCG.2010.199.

Ropinski, T., Oeltze, S., Preim, B., 2011. Survey of glyph-based visualization techniques for spatial multivariate medical data. Comput. Graph. 35 (April (2)), 392–401 pages.

Li, Y.N., Li, D.J., Zhang, K., 2015. Metaphoric transfer effect in information visualization using glyphs. In: Proceedings of the 8th International Symposium on Visual Information Communication and Interaction (VINCI '15). ACM, New York, NY, USA, pp. 121–130. http://dx.doi.org/10.1145/2801040.2801062.



Dr. Daniel Simões Lopes has a degree in biomedical engineering from the University of Lisbon and graduated in computational engineering under the framework of the UT Austin|Portugal Program. Currently, he is a postdoc researcher and Head of Biomedical Research at the Visualization and Intelligent Multimodal Interfaces Group at INESC-ID Lisboa, where he carries research and development in several topics, namely, collision detection, graphical user interfaces, virtual reality, computer-aided design, and interactive visualization of 3D medical images.



Rafael Kuffner dos Anjos is a Computer Science and Engineering PhD student at Técnico, Universidade de Lisboa. His doctoral studies focus mostly on Video-based Rendering. As member of the BlackBox project (sponsored by the European Research Council), he develops research on document performance composition by proposing new approaches on information representation of human performers. Other interests include 3D reconstruction techniques, data capture, and image-based collision detection. He was also a research member of the Global Lab at the National Institute of Informatics in Tokyo, being a part of the iCO2 project.

Joaquim Jorge received his PhD from Rensselaer Polytechnic Institute in 1995, coordinates the VIMMI research group at INESC-ID and is Full Professor of Computer Graphics and Multimedia at Técnico, Universidade de Lisboa. He is Editor-in-Chief of the Computers & Graphics Journal (Elsevier), a Fellow of the Eurographics Association, Senior Member of ACM and IEEE, serves on the ACM Europe Council and Chairs the ACM/SIGGGRAPH Specialized Conferences Committee. He organized 35 + international scientific events, was Eurographics 2016 papers co-chair, served on 180 + program committees and (co)authored 290 + publications in international peer-refereed venues. His research interests include multimodal user interfaces, 3D visualization and modeling.