

An Algebraic Process for Visualization Design

Gordon Kindlmann and Carlos Scheidegger

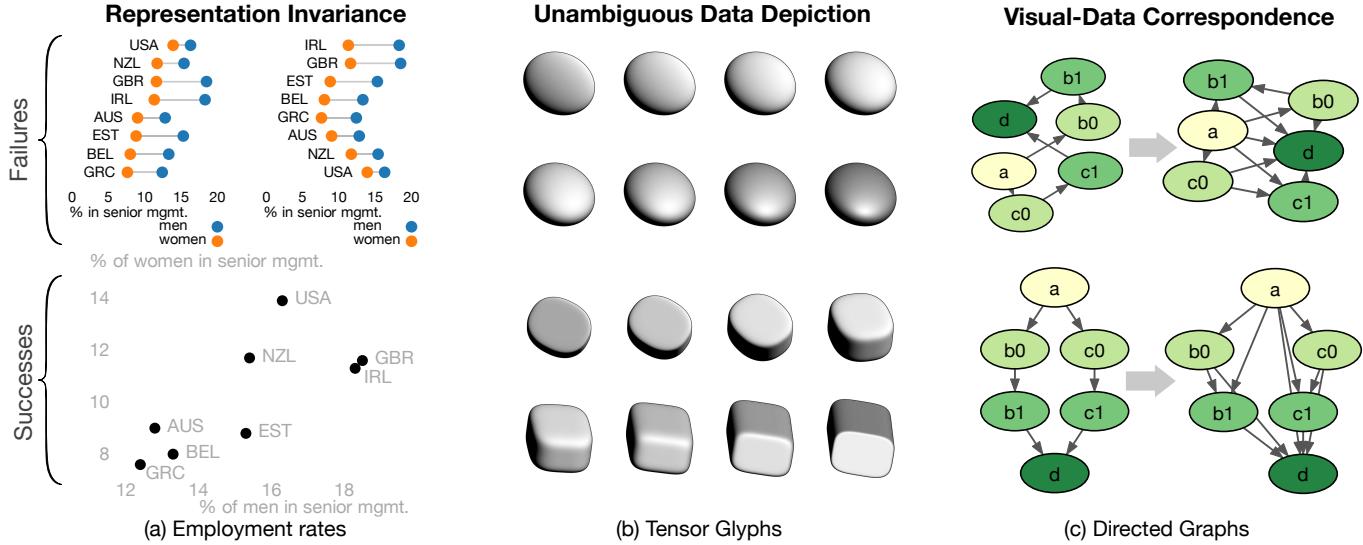


Fig. 1: Examples of violating (top) and respecting (bottom) our three proposed visualization design principles. The Principle of Representation Invariance says data, not spurious details of its representation, should determine the impression of the visualization. Permuting the countries in (a) creates different impressions in the top visualizations, but not in the bottom scatterplot. Visualizations that respect the Principle of Unambiguous Data Depiction make large changes in the data clearly visible. To visualize tensors (b), bas-relief ambiguity makes ellipsoid glyphs for different tensors appear similar (top); superquadric glyphs avoid the problem (bottom). With the Principle of Visual-Data Correspondence, meaningful changes in data should produce analogously meaningful changes in impressions, according to the mathematical structure of both data and visual spaces. In (c) the directed graph’s partial order is unchanged by adding some edges. The force-directed placement (top) is completely re-arranged, but not a layout indicating node rank with vertical position (bottom). All principles are consequences of Equation (1), the basis of our algebraic design process.

Abstract— We present a model of visualization design based on algebraic considerations of the visualization process. The model helps characterize visual encodings, guide their design, evaluate their effectiveness, and highlight their shortcomings. The model has three components: the underlying mathematical structure of the data or object being visualized, the concrete representation of the data in a computer, and (to the extent possible) a mathematical description of how humans perceive the visualization. Because we believe the value of our model lies in its practical application, we propose three general principles for good visualization design. We work through a collection of examples where our model helps explain the known properties of existing visualization methods, both good and not-so-good, as well as suggesting some novel methods. We describe how to use the model alongside experimental user studies, since it can help frame experiment outcomes in an actionable manner. Exploring the implications and applications of our model and its design principles should provide many directions for future visualization research.

Index Terms—Visualization Design, Symmetries, Visualization Theory

1 INTRODUCTION

Visualization research has made dramatic discoveries: faster algorithms, for showing more data, and more patterns within data, in ways that are more finely attuned to users’ needs and abilities. The resulting visualization tools play a vital role in business and science. Visualization research is continuously energized by new data types, new analysis tasks, and new applications.

Theoretical visualization research builds a rigorous foundation for

the field by studying fundamental properties of how visualizations work. Such research may draw on psychological studies of vision and cognition, but within the visualization community, theory papers often categorize data, tasks, and methods into various taxonomies. These are invaluable for understanding the landscape of our field, but descriptive taxonomies offer limited insight into how to design new visualizations, especially in the face of new data types, tasks, and applications.

Practitioners may find some guidance in the illustrative examples of good and bad visualization design, such as those gathered by Tufte. Tufte offers maxims like “maximize the data-ink ratio” and “show data variation not design variation” [48]. With respect for Tufte’s eloquence and impact, we feel such maxims work better as summaries of what good visualizations have in common, than as concrete guidance for how to create good visualizations, or improve bad ones.

Our work strives to be both theoretical and constructive. We present an algebraic basis for designing data visualizations. We give practical guidance through the three design principles illustrated by the examples in Figs. 1 through 4, but we aim for generality by describing visual-

• Gordon Kindlmann is with the University of Chicago. gk@uchicago.edu.
 • Carlos Scheidegger is with the University of Arizona, though this work was completed while at AT&T Labs. cscheid@cscheid.net.
 • The two authors contributed equally to this work.

Manuscript received 31 Mar. 2014; accepted 1 Aug. 2014; date of publication xx xxx 2014; date of current version xx xxx 2014.

For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

ization design in mathematical terms. We recognize three fundamental elements of creating and viewing visualizations:

- **Mathematical structure in the underlying data.** Data comes in as many forms as there are things to model, measure, and describe. We can distinguish different forms of data at an abstract level by characterizing the mathematical types of the individual data points, their organization, and their operational context, which we collectively call “structure”. For hierarchies like trees, the structures includes the partial order representing “is-children”, and order-preserving maps between trees. For vectors in Euclidean space, the structure includes not only the length and direction of the vectors themselves, but the relevant linear transformations on the space and their properties. The structure that matters for a given data source will likely vary according to the goals and application of the visualization.
- **Concrete representation of data in a computer.** Visualization programs start with computational representations, but representations have properties not intrinsic to the data. A set of data points has no intrinsic ordering, but representing it as a list or table forces one arbitrary ordering. Eigenvectors are represented with vectors, with a necessary but arbitrarily choice between \mathbf{v} and $-\mathbf{v}$. In distinguishing between data and representation, we also include more complex processes where an underlying object of interest to be visualized (the “data”) can be accessed only through samples of it (the “representation”); this includes statistical samples of a population, or samples on a discrete grid of an underlying continuous field.
- **Mathematical structure in the perception of visualizations.** The human visual system processes visualizations and their constituent visual encodings to form an impression of the visualization. Psychology has produced mathematical descriptions of visual processing. Opponent color theory, for example, defines for every color an opponent color that yields gray upon their combination. This endows perceptual color space with a negation operation.

We propose three visualization design principles in terms of these elements, and we offer evocative names for failures of the principles.

- The Principle of Representation Invariance (or just Invariance) says that visualizations should be invariant with respect to the choice of data representation: changing the representation should not change the visualization. A visualization failing this principle has a *hallucinator*: a different impression was created (hallucinated, in fact) out of nothing but a different representation of the same data.
- The Principle of Unambiguous Data Depiction (or just Unambiguity) says that visualizations should be unambiguous: changing the underlying data should produce a change in the resulting visualization. Failing this principle, a visualization has *confusers*: changes in the data that are effectively invisible to the viewer of the visualization.
- The Principle of Visual-Data Correspondence (or just Correspondence) says that significant changes in the data should meaningfully correspond with noticeable changes in the visual impression and vice versa. If an important change in data is not clearly manifested in the visualization, it has *jumbled* the data. If a clear and obvious transformation of the visualization corresponds with an unimportant change in the data, the visualization is *misleading*.

We intend the principles and their failures to be understood intuitively, but they are grounded in an algebraic model, presented in Section 3. Section 4 illustrates the principles with examples from Figs. 1 through 4 and the literature. Section 5 shows how our approach facilitates iterative and methodical examination and improvement of a visualization, through the redesign of a visualization of the employment gender gap in senior management across different countries. Finally, Sections 6 and 7 suggest ways in which our proposal can be generalized, and how it accounts for research advances (and current discussions) in multiple areas of visualization.

2 RELATED WORK

As our general goal is to connect the structure of visualized data with the structure of the perception of visualizations, the related work spans data types, perception, and previous taxonomies and theories that con-

nnect them. Bertin’s seminal work enumerates a set of data types and good “retinal variables” [2], which Mackinlay’s APT [28] encodes and systematizes, which in turn drive automatic visualization generation in Tableau [29]. Ongoing theoretical visualization work organizes data, tasks, and methods into taxonomies [44, 46, 5].

The traditional base data types in visualization (i.e. categorical or nominal, ordinal, interval, and ratio) are typically justified with reference to Stevens [45, 56]. Less often discussed is how Stevens uses the mathematics of group symmetry to give those types precise meaning. For example, any permutation of the labels of categorical data is an equivalent labeling, thus the group (in the abstract algebra sense) of permutations contain the symmetries of categorical data. Ordinal, interval, and ratio data have symmetry groups of monotonically increasing, affine ($f(x) = mx + b$), and linear ($f(x) = mx$) functions, respectively. Stevens’s mathematical approach directly inspires ours: we use “symmetry” to refer generally to invertible transformations that, when applied to each in a set of things, map back to the same set. Our “data symmetry” and “visualization symmetry” terms are based on this. Stevens’s motivation was to identify the “permissible” or “invariantive” statistics on each measure. With ordinal data, for example, one can meaningfully compute a median because it is invariant to monotonic transforms, while the mean is not. Our Invariance Principle expresses the same idea: a visualization should not depend on the specifics of how the underlying data is represented.

Previous visualization research has addressed human perception in different ways. Much work is organized around low-level properties of the human visual system originally described by Cleveland and McGill [11]. Wattenberg and Fisher make Gestalt models of visualization perception by relating the scale-space structure of the visualization stimuli to structure at different scales in the underlying data [57], which is a psychology-based instantiation of our Correspondence Principle. The high-level cognitive processes used for interactive visualization are also studied, for example in the contexts of distributed cognition [26] or narrative story-telling [21]. A more concrete mechanism for how we relate to visualizations is the *affordance*, originally described by Gibson as something perceived in the environment to provide the possibility of action [12]. Ware generalizes this to include properties of visualizations and interfaces, arguing that thinking in terms of affordances helps design better visualizations [56]. We embrace this strategy, and aim to implement it with mathematics. Specifically, our algebraic model describes how visual affordances should match important low-level tasks, according to our Correspondence Principle.

Other previous theories of the visualization process have used mathematical descriptions. Ziemkiewicz and Kosara highlight how injectivity, surjectivity and bijectivity are important properties of visualizations [63]. Our model uses some similar ideas (in particular their injectivity is analogous to our Unambiguity Principle), but we focus on how visualizations are perceived generally, not how they are “read” as a sequence of syntactic units. Bar charts, for example, require an arbitrary choice in ordering the bars, each of which is equally surjective in their view. We see this as an Invariance failure, however, since the different orderings are perceptually different (Sect. 5). We also see the space of data and of visual stimuli as supporting a richer description than that of mere sets; our Correspondence Principle seeks to match mathematical structure in data with that in visual perception.

Demiralp et al. suggest matching distance functions (in data space and in perceptual space) as a unifying principle for designing and evaluating visualizations [8]. We share interest in this promising approach, which is essentially a quantitative specialization of our Correspondence Principle. The descriptive power of distance functions and metrics does have limits, however. Partial orders, for example, are not symmetric, so they cannot be represented in a metric space. Psychologists know that judgments of similarity between objects can also be asymmetric [50]. We see our work as providing a mechanism for investigating which structures in the data are preserved and reflected in the visualization with more generality than allowed by metric spaces.

Mackinlay’s original automated visualization tool is formulated in terms of “Expressiveness” and “Effectiveness” criteria for evaluating graphical languages [28]. Expressive visualizations convey all the

Principle Name	Precondition	Requirement	Name for failure	Failure definition
Representation Invariance	$\alpha = 1_D$	$\omega = 1_V$	Hallucinator	$H(v) = \{h \mid r_2 = h \circ r_1 \text{ and } v \circ h \neq v\}$, over all representations r_1, r_2
Unambiguous Data Depiction	$\omega = 1_V$	$\alpha = 1_D$	Confuser	$C(v) = \{\alpha \mid v \circ r \circ \alpha = v \circ r\}$
Visual-Data Correspondence	$\alpha \neq 1_D, \omega \neq 1_V$	$\alpha \cong \omega$	Jumbler and Misleader	(see Sec.3.3 text)

Table 1: Algebraic visualization design principles for evaluating a visualization method v , expressed in terms of (1).

properties (or “facts”) of a given dataset, and nothing more. Visualizations that are not expressive because they fail to show important structure in the data will violate our Unambiguity Principle. Failing to be expressive by suggesting non-existent facts about the data can arise either by violating our Invariance Principle (showing accidental properties of the representation rather than the data), or by violating our Correspondence Principle with a misleader. For Mackinlay, *effective* visualizations match important data attributes to readily perceived visual attributes. If a visualization violates our Correspondence Principle with a jumbler, it will likely also fail to be effective, but effectiveness failures can also arise from violating our Unambiguity Principle (when important changes in data are mapped to perturbations in the visualization too subtle to make an impression).

Studying explanatory graphics in cognitive psychology, Tversky et al. propose a Congruence Principle: the visual (external) structure of an explanatory graphic should correspond to the desired structure of the mental (internal) representation in the viewer [52, 51]. Our Correspondence Principle essentially restates this in the context of data visualization, but with more mathematical descriptions of how *changes* in data relate to changes in the visualization, and of how visualizations may fail. Tversky et al. also propose an Apprehension Principle [52] by which graphics should be readily and accurately perceived and comprehended. Like Mackinlay’s Effectiveness criteria, this relates to both our Unambiguity and Correspondence principles.

In terms of Munzner’s nested model for visualization [33], our model allows designers to reason about which (and how) structures from the data/operation abstraction layer will be reflected in the encoding/interaction technique layer. In this way, our theory can predict which visualizations will be good for which tasks. Relative to Meyer et al.’s extended nested model [31], our theory suggests mathematical guidelines for the abstraction layer, directly from the structure of the data. As a trivial case, we could say that if a visual encoding preserves the metric structure of the original dataset, then it will be good for assessing similarities between input points.

3 THEORETICAL MODEL AND DESIGN PRINCIPLES

This section states our fundamental equation of algebraic visualization design after defining some terms and variables, then examines the equation structure with a *commutative diagram* [10], and finally uses the diagram to organize the statement of our three design principles.

Our algebraic model describes relationships between the three elements of visualization identified in Section 1: the space of data to be visualized (denoted D), the space of data representations (R), and the space of visualizations (V). The mappings between these spaces are captured in a single equation, shown with its commutative diagram:

$$\begin{array}{ccccc} & D & \xrightarrow{r_1} & R & \xrightarrow{v} V \\ v \circ r_2 \circ \alpha & = & \omega \circ v \circ r_1 & & \\ \alpha \downarrow & & \downarrow & & \omega \downarrow \\ D & \xrightarrow{r_2} & R & \xrightarrow{v} & V \end{array} \quad (1)$$

Note that lowercase r denotes a mapping from data D to representation R , and lowercase v denotes a visualization method mapping from R to visual stimulus V . We view visualizations as acting not on data itself but on some concrete representation in the computer. The mappings from D back to D (or mappings on D), are termed *data symmetries*, denoted α . The *visualization symmetries*, denoted ω , are mappings on V . We refer to data symmetries α and visualization symmetries ω frequently throughout this paper, so we use α and ω for brevity. We chose alpha and omega to emphasize how these mappings relate to the

beginning (the data) and to the end (the visual stimulus), respectively, of the visualization process. The identity mapping (sending each point in a space back to itself) on D and on V is denoted 1_D and 1_V , respectively.

In a commutative diagram, when two nodes are connected by two paths, the composition of functions along either path must be the same. The equality in (1) is between two possible paths from the upper-left D to the lower-right V . Going down then right is $D \xrightarrow{\alpha} D \xrightarrow{r_2} R \xrightarrow{v} V$, or in terms of function composition (read right to left) $v \circ r_2 \circ \alpha$, and going right then down is $D \xrightarrow{r_1} R \xrightarrow{v} V \xrightarrow{\omega} V$ (or $\omega \circ v \circ r_1$).

Using our algebraic design model requires finding and understanding the possible (α, ω) pairs that make the diagram commute; each such (α, ω) gives a different solution to (1). In successful designs, the α represent relevant data properties and low-level tasks, and they are matched via (1) with ω that align with perceptual channels or visual affordances. Visualization designers can start with an α and investigating the resulting ω . For a given data symmetry α , there is always *some* visual symmetry ω that solves (1). Specifically, for each data symmetry α on D and x in D , computationally represented by $r_1(x)$ and visualized by $v(r_1(x))$, we can define $\omega(v(r_1(x))) = (\omega \circ v \circ r_1)(x)$ as $v(r_2(\alpha(x))) = (v \circ r_2 \circ \alpha)(x)$: the visualization of the representation $r_2(\alpha(x))$ of the transformed data $\alpha(x)$. An ω may act locally to change location of a dot or the color of a circle, or it may act more globally, replacing every color in a colormapped field with another color.

We use (1) to define three algebraic visualization design principles, and to characterize ways in which a visualization may fail the principles. Table 1 summarizes these for reference.

3.1 Representation Invariance: $\alpha = 1_D \Rightarrow \omega = 1_V$

Defining our Principle of Representation Invariance starts by setting α to the identity mapping 1_D on the data space D , which collapses the left side of the commutative diagram, so that we are considering only a single dataset. Then the right side should also collapse (that is, $\omega = 1_V$): the visualization should be the same even from two different representations $r_1 \neq r_2$. If not, then changes in representation can lead to changes in the visualization, or, a single dataset maps to many distinct visualizations. The set of erroneously consequential changes in representation are the *hallucinator* $H(v)$ of a visualization v . Invariance is satisfied when the hallucinator is empty: $H(v) = \emptyset$.

3.2 Unambiguous Data Depiction: $\omega = 1_V \Rightarrow \alpha = 1_D$

The Unambiguous Data Depiction Principle is the converse of Invariance. To consider a single visualization and ask how it arose, we collapse the right side of the commutative diagram by setting $\omega = 1_V$. Then the left side should also collapse (that is, $\alpha = 1_D$). If it does not, it means there is some significant change $\alpha \neq 1_D$ in the data which leaves the visualization unchanged, or, many datasets map to a single visualization. Changes in the data that are invisible to the viewer of a visualization v are the *confuser* $C(v)$. The Unambiguity Principle is satisfied when the confuser is the identity on data: $C(v) = \{1_D\}$. Our definitions of Unambiguity and the confuser in Table 1 assume that Invariance is satisfied: erroneously consequential changes in representation could confound the structure of the ambiguity.

3.3 Visual-Data Correspondence: $\alpha \cong \omega$

The Invariance and Unambiguity Principles are concerned with solutions to (1) where either the data symmetry α or the visualization symmetry ω is the identity. Assuming both Invariance and Unambiguity hold, the Correspondence Principle is satisfied when *neither* α nor

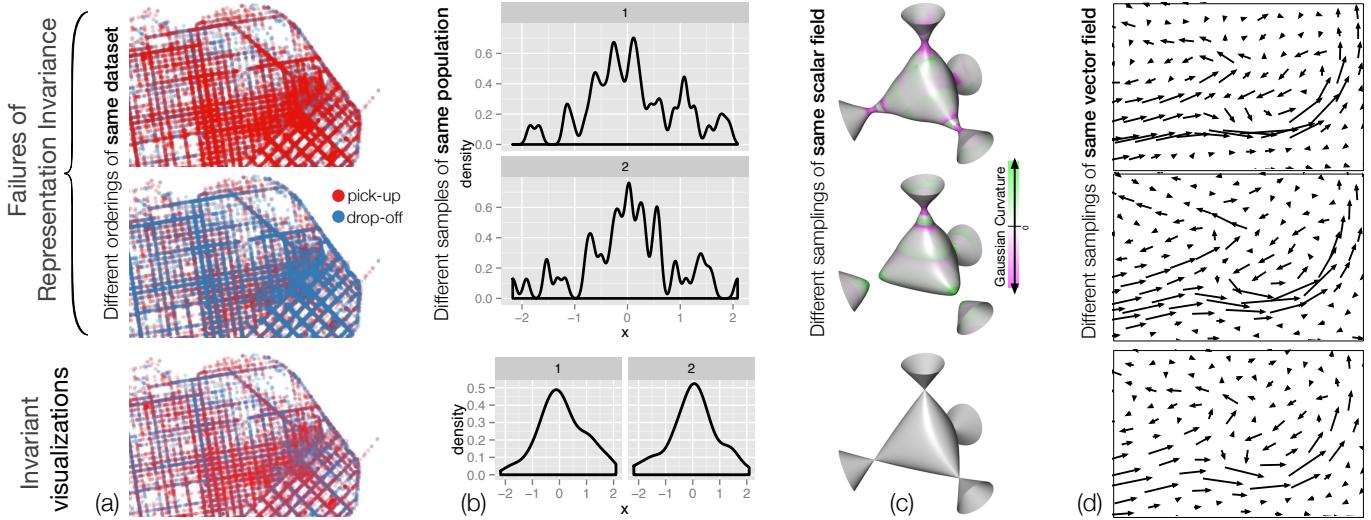


Fig. 2: Our Invariance Principle illustrated with taxi pick-ups and drop-offs (a), two different samples from a population (b), volume renderings of sampled 3D cubic polynomial (c), and vector glyphs in a 2D flow field (d). The upper pair of adjacent visualizations are of exactly the same underlying data or object, but give different impressions due to arbitrary differences in representation, sometimes beyond the control of the designer. The bottom row demonstrates the Invariance Principle with visualizations that do not depend on representation choice.

ω is the identity, and they solve (1) in a particular way. We informally note this “ $\alpha \cong \omega$ ” to convey the desired congruence [52] between the data and visualization symmetries. Applying the Correspondence Principle proceeds by defining a particular symmetry of interest (either on the data space or on the visual space), finding the symmetry on the other side of the commutative diagram that solves (1), and then assessing whether there is a reasonable correspondence between α and ω . More so than the other two, the Correspondence Principle implements the task-dependence of visualization design, since designers can choose α to model data properties of particular value for some application. Choosing the visualization symmetry ω is informed by the current scientific understanding of the visual system and our basic mechanisms of visual comprehension. Cleveland and McGill famously determined, for example, that changes in positions along a common scale are better distinguished than changes in length, which are in turn better distinguished than changes in area or color saturation [11].

We suggest that two terms are appropriate for describing Correspondence failures, according to whether one is judging the data symmetry relative to the visual symmetry or vice versa. We acknowledge the distinction between the two may not be clear cut, and that the mathematical vocabulary for their expression is currently lacking. When we pick a meaningful data symmetry α and find the matching visual symmetry ω unsatisfactory, we say the visualization has a *jumbler*, or that it jumbled α . When a clear and readily apparent ω turns out to correspond to a complicated or inconsequential α , we say the visualization has a *misleader*, since ω gave a misleading interpretation of the data. If an obvious ω maps to a visual stimulus that the visualization v can never produce from any representation, then there is no α that solves (1); this is an additional kind of misleader.

4 EXPLAINING PRINCIPLES AND FAILURES BY EXAMPLE

This section describes the principles and their failures by reference to the examples in Figs. 1, 2, 3, 4, and the literature. We hope to convey that our principles are not a new set of *rules* to obey, but are *tools* for investigating and describing how a visualization does or does not depict data, and for improving visualizations in an informed way.

4.1 Representation Invariance

Figures 1 and 2 show examples of hallucinators. The hallucinators in Figs. 1(a) and 2(a) are permutations of a list representing a set. Considered as samples from a population, sets of countries (or taxi pick-up and drop-off locations) have no intrinsic ordering, but representing the set as a list necessarily picks some order. Note that in Fig. 1(a),

the scatterplot does not have a hallucinator, while in Fig. 2(a), the scatterplot does. This happens because the hallucinator in Fig. 2(a) arises from the non-commutativity of the “over” operator when different colors overdraw each other, even with low opacity [37].

Figure 2(b) illustrates an Invariance Principle failure that may occur in statistical visualization. Two samples drawn from a single underlying population are plotted with the hope of getting a sense of that population. But the most visually prominent differences in peaks and valleys in the top two plots are features of the sample (the representation), not of the population (the data). In the bottom of Fig. 2(b), the same shape of the underlying population is revealed from both samples by the application of kernel density estimation [35] and an appropriate bandwidth (Gaussian kernel width), producing two plots that can be considered visually equivalent (i.e. $\omega = 1_V$).

Figures 2(c) and (d) have hallucinators associated with the different grids that sample the same underlying continuous field. The volume renderings in Fig. 2(c) depict the Cayley cubic polynomial $f(x, y, z) = x^2 + y^2 - zx^2 + zy^2 + z^2 - 1$, for which isosurface $f(x, y, z) = 0$ has zero Gaussian curvature [58]. Renderings with the Catmull-Rom filter, with insufficient continuity and accuracy (in the Taylor-series sense), visually emphasize the sampling grid and show shapes with non-zero Gaussian curvature. Möller et al. design filters (for field reconstruction by convolution) of arbitrary continuity and accuracy, including those that can exactly reconstruct cubics [32]. A better filter removes the hallucinator because it leads to the same (correct) rendering regardless of sampling grid orientation. The hedgehog plots in the top of Fig. 2(d) visualize individual vector values with arrows at every regular grid point. This may be good for data quality inspection, but the impression of the smooth underlying flow pattern is unduly affected by the sampling grid. The hallucinator is removed with a glyph placement strategy based on reconstructing the underlying continuous vector field, such as the image-guided placement of Turk and Banks [49].

4.2 Unambiguous Data Depiction

Recognizing and characterizing confusers (failures of Unambiguity) provides an actionable path for visualization design. The ellipsoids in the top of Fig. 1(b) visualize very different tensors, but the glyphs appear very similar due to bas-relief ambiguity, for which the confuser is scaling along the view direction [1]. Though it preceded our current terminology and theory, the same confuser was used in designing superquadric tensor glyphs [23]. Subsequent work generalized superquadric glyphs to a larger class of tensors, based on design principles analogous to those of this paper [40]. Confusers also help

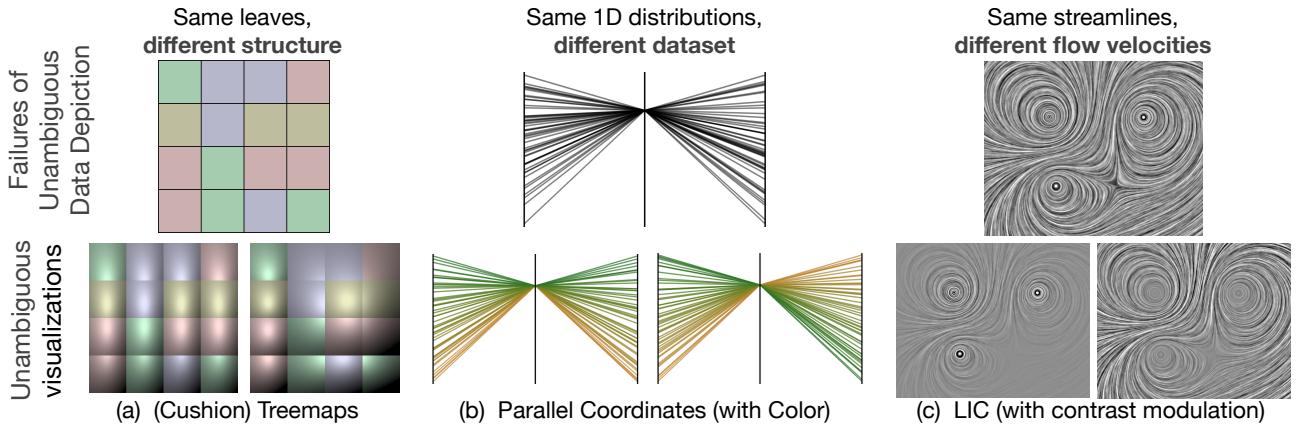


Fig. 3: Demonstrating the Principle of Unambiguous Data Depiction with ambiguous visualizations in the upper row, and their disambiguations in the lower row. Standard treemaps (a) do not clearly show hierarchy; cushion treemaps do [54]. Parallel coordinate plots (b) are ambiguous when more than one point shares a coordinate. Line color can disambiguate (with a risk of Invariance failure). LIC (c) uses streamlines computed on a normalized-velocity field. Modulating LIC contrast with local velocity reveals otherwise unseen velocity variation patterns.

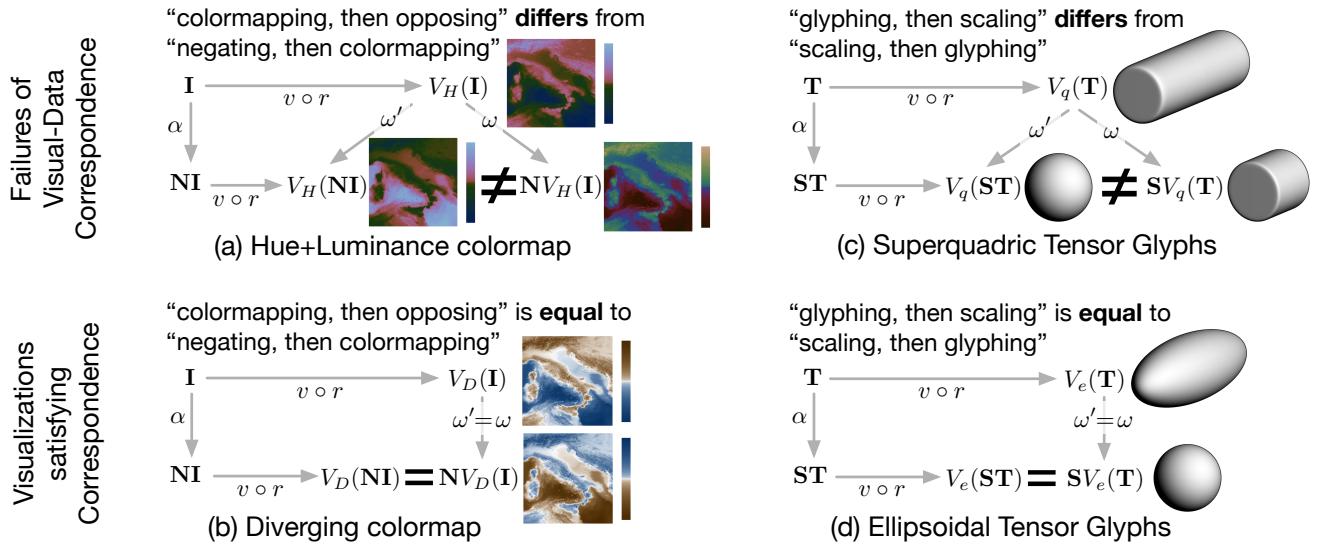


Fig. 4: The Correspondence Principle facilitates visual interpretation of data by ensuring that changes in data cause changes in the visualization that are meaningfully aligned with perceptual channels or visual affordances. In (a), the negation α of elevation prior to colormapping induces change ω' in the visualization, but “negating” colors (switching with opponent hues) makes a very different change ω . With a diverging colormap (b), ω' and ω are the same, enabling a simple visual interpretation of negation. Even though superquadric tensor glyphs overcome the bas-relief ambiguity of ellipsoidal glyphs, they do not permit a straightforward visual interpretation of scaling along an eigenvector. Scaling the superquadric glyph (c) creates a different change ω than the change ω' induced by visualizing the scaled tensor. With ellipsoid glyphs (d), they are the same.

characterize another development in tensor visualization. The standard diffusion tensor colormap assigns $(R, G, B) = (|x|, |y|, |z|)$ to principal eigenvector $\mathbf{v} = (x, y, z)$ [36]. This scheme appropriately assigns the same color to both representations \mathbf{v} and $-\mathbf{v}$ of a single eigenvector. But there is a confuser: all eight distinct vectors $(\pm x, \pm y, \pm z)$ map to the same color. Recognizing that identifying antipodal points \mathbf{v} and $-\mathbf{v}$ on the sphere produces the real projective plane \mathbf{RP}^2 , Demiralp et al. propose eigenvector coloring by embedding a standard parameterization of \mathbf{RP}^2 , the Boy’s surface, into 3D perceptual colorspace [7]. This replaces the eight-fold ambiguity with a negligible ambiguity at the (zero-measure) self-intersection in the Boy’s surface.

The treemap in Fig. 3(a) (top) shows sixteen nodes, colored by type, in a tree structure [43]. In this worst-case scenario, the coincidental horizontal and vertical lines create a large confuser. The viewer cannot determine the depth of the hierarchy, let alone sibling relationships. Van Wijk and van de Wetering resolve this ambiguity with cushion treemaps [54]. The differences in abstract tree structure becomes readily perceived when rendered as directionally shaded surfaces, subdivided according to the hierarchy (Fig. 3(a) bottom).

Standard parallel coordinate plots (Fig. 3(b)) have a large confuser. Specifically, if two data points share one coordinate value, there is an α that switches the value of a neighboring coordinate between the two points, yet gives the same resulting visualization. Assigning to each data point a distinguishing feature, such as color (Fig. 3(b) bottom), removes the confuser. Straightforward fixes, however, may lead to Invariance failures. The order of data points, which determines color, is now a hallucinator. Previous work uses curves instead of lines to connect parallel coordinates, which also removes the confuser [13].

Figure 3(c) depicts flow by Line Integral Convolution (LIC), which convolves an underlying noise texture with streamlines computed with normalized velocity [6]. Changing flow magnitude while preserving direction is a confuser since the computed streamlines will be the same. Modulating the LIC contrast with a monotonic function of vector magnitude (Fig. 3(c) bottom) removes the confuser, as does convolving with multi-frequency noise textures [24]. Though generally unwelcome, confusers can be ignored in certain application areas. Vector field topology, for example, creates depictions that are invariant to changes in flow magnitude [17], in which case this confuser is intentional.

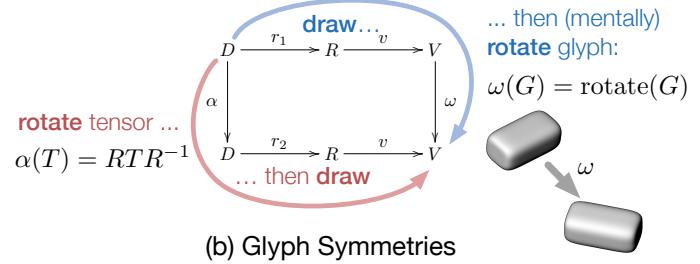
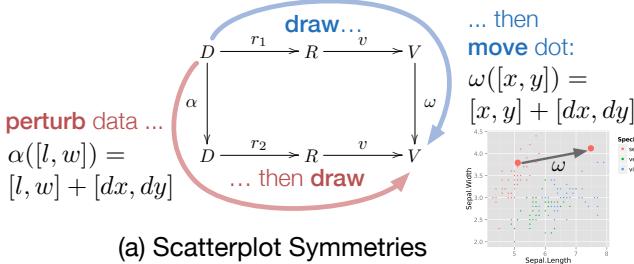


Fig. 5: Correspondence Principle (α, ω) symmetry pairs. Correspondence asks that significant changes α in data are meaningfully matched with significant changes ω in the visual stimulus. In (a), translating one scatterplot dot is directly matched to an additive update of the corresponding data point. In (b), a mental rotation of the glyph shape is matched with a mathematical rotation of the tensor.

4.3 Visual-Data Correspondence

The design of an elemental visualization ingredient, the arrow, can be understood in terms of our Correspondence Principle. Just as vectors v and $-v$ are mathematically related by negation, we expect glyphs visualizing these vectors (such as \rightarrow and \leftarrow) to also be perceptually related by something akin to negation or flipping. In fact, in considering a perceptual theory of flow visualization, Ware describes how arrow shape and arrangement may be designed to trigger asymmetric responses in particular neurons in our visual cortex [55].

Failures of the Correspondence Principle are relative to particular data or visual transformations. For example, the visualizations of directed acyclic graphs in Fig. 1(c) use color to indicate node rank. The data symmetry α of interest here is adding redundant (rank-preserving) edges to the graph. Force-directed node placement (Fig. 1(c) top) fails the Correspondence Principle with this α because the layout changes significantly (despite the rank being unchanged), with a complex visual symmetry ω . Directly visualizing rank with vertical position (Fig. 1(c) bottom) solves this problem, since adding the edges cannot change the layout significantly.

Figs. 4(a) and (b) illustrate the Correspondence Principle with two different colormaps of elevation x relative to sea level. An interesting data symmetry α is negation $\alpha(x) = -x$; it preserves the coastline and flips being above or below water. Based on opponent color theory [61], we also define visual symmetry ω as mapping each hue to its opponent hue (i.e. mapping blue to yellow and vice versa, while preserving luminance). A colormap satisfies the Correspondence Principle when the two kinds of negation in fact correspond, i.e., α and ω satisfy our design equation (1). With the hue-and-luminance colormap in Fig. 4(a), negating elevation and then colormapping creates (via (1)) a complicated visual symmetry ω' very different than the intended ω . But with the diverging colormap in Fig. 4(b), the symmetry ω' found by negating then colormapping is the *same* as the symmetry ω that “negates” hues. This (α, ω) pair solves (1) and Correspondence is satisfied. Our theory is providing a precise mathematical demonstration of what had been known from experience about designing diverging colormaps: opponent hues offer a visual affordance for negating values [56]. In particular, grays are particularly good for depicting zero because if $v(0) = \text{gray}$ and $\alpha(0) = 0$, then $\omega(\text{gray}) = \text{gray}$.

The data symmetry α in Fig. 4(c) and (d) is scaling along a single eigenvector, specifically scaling a large eigenvalue down to match two smaller and equal eigenvalues. Although better than ellipsoids relative to the Unambiguity Principle (Fig. 1(b)), the superquadric glyph in Fig. 4(c) fails Correspondence for this scaling α . The symmetry ω' that solves (1) is not the symmetry ω that intuitively corresponds to α : scaling the glyph by the same amount as α along the same eigenvector. Ellipsoid glyphs (Fig. 4(d)), however, provide a visual affordance that superquadrics lack: to see a scaled glyph and interpret it as a scaled tensor. But the Correspondence failure is α -specific; for a different α of possible interest, scaling equally along all eigenvectors, both ellipsoid and superquadric glyphs satisfy Correspondence.

Figure 5 shows two more Correspondence examples. Fig. 5(a) highlights a single data point in a scatterplot, with which we define a visualization symmetry ω of interest: translating the point to somewhere else in the visualization. Applying Correspondence amounts to

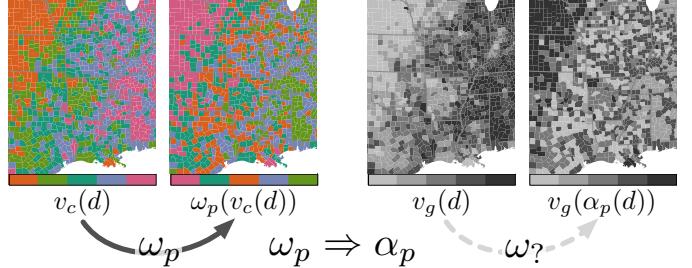


Fig. 6: A Correspondence Principle misleader of ordinal data. A collection of hues appropriate for colormapping categorical data permits a visual symmetry ω_p of permutation (left). This is misleading, since the corresponding data permutation α_p is not actually a symmetry on the category of ordinal data. This is evident with a grayscale colormap suited for ordinal data (right).

asking, “what change α in data corresponds to moving this point?” In a scatterplot, α is simply an additive coordinate change. In a principal component analysis (PCA), however, the α is harder to reason about, since it involves the rotation induced by PCA. For this particular ω , then, we say that scatterplots better conform to the Correspondence Principle than does PCA, and that PCA has a misleader.

Note that solving (1) with translations α and ω includes a larger class of visualization designs than scatterplots. For example, Trumbo’s “Rows and Columns” principle for bivariate colormaps preserves the univariate components by encoding each axis with an independent perceptual color channel [47]. The α and ω involved are then translations in the 2-D data space and in a 2-D slice of perceptual color space.

In Fig. 5(b), a tensor glyph is shown with a shaded 3D shape. A data symmetry α that rotates the tensor corresponds with a visual symmetry ω of mentally rotating the glyph geometry. The use of such a visual symmetry is justified by knowledge of our abilities to perform mental rotations [42] and to perceive shape from shading [20].

Most of the Correspondence failures in previous figures are jumblers. Figure 6 shows a misleader in displaying unemployment data binned to create ordinal data. A collection of roughly isoluminant hues (appropriate for categorical data [15]) affords the visual symmetry of permutation: each image in the left pair creates an equivalent impression, and the visual symmetry ω_p maps between them. Solving (1) finds the corresponding data transform α_p , but permutation is not a symmetry of ordinal data [45], as is obvious with a more appropriate grayscale colormap (Fig. 6 right). The isoluminant hue colormap misled us into thinking the data was symmetric under permutation.

5 A WORKED EXAMPLE

We use our theory to reason about a sequence of design improvements in a simple visualization, shown from left to right in Fig. 7. We start with Jonathan Schwabish’s study and redesign [41] of a visualization by Catherine Rampell in the New York Times [39] of the percentage of men versus women employed as senior managers in various countries. Design #1 is due to Rampell, and Designs #2 and #3 are due to Schwabish. Within the rest of this section, “rate” always refers to

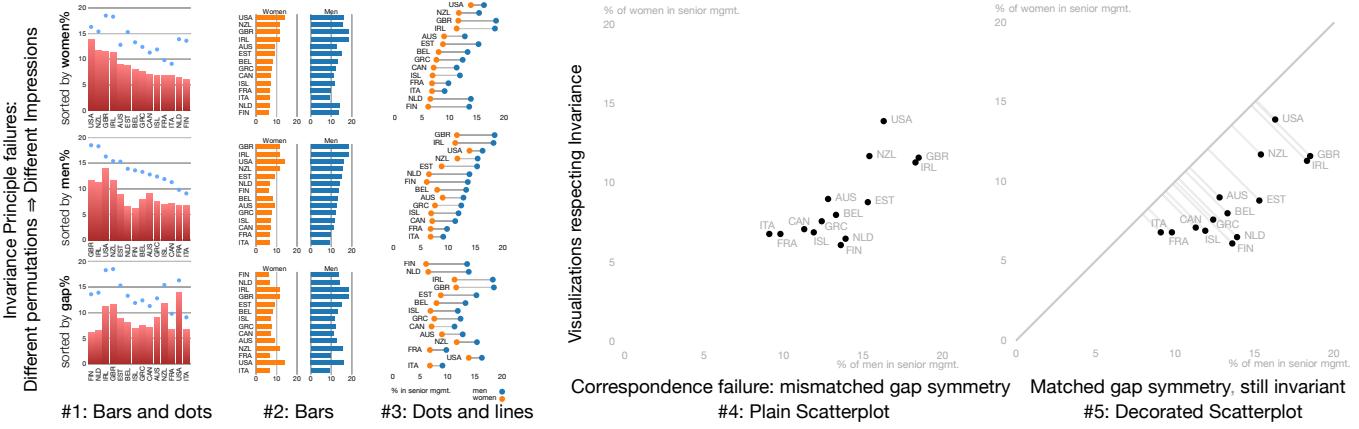


Fig. 7: Visualizations of gender gaps (in rates of employment as senior managers) in designs #1, #2, and #3 fail the Invariance Principle by sensitivity to permutations of countries. The scatterplot in #4 respects Invariance but lacks a clear (in the sense of Cleveland and McGill) visual symmetry for the important *employment gap* data symmetry. This is remedied with annotations in the decorated scatterplot #5.

“rate of employment as senior management”. In Design #1, red bars show the female rate, while blue dots show the male rate. For clarity in our discussion, we use only the 14 countries with highest percentage women employed in 2008 as senior managers.

5.1 Data symmetries model low-level tasks

We start on the left side of the commutative diagram of (1) with *data symmetries*, the transformations α of data space. The α model questions and properties of the data space important for the visualization. We think of these as a formal implementation of “low-level abstract” operations in Munzner’s nested model [33], or like the “manipulate” nodes in Brehmer and Munzner’s recent typology [5]. We describe each symmetry α first with words and then with equations, in which x_M and x_W denote the rates for men and women, respectively.

1. α_1 : What if the rate was different for just one gender?
Either $x'_W = x_W + k$ and $x'_M = x_M$, or, $x'_M = x_M + k$ and $x'_W = x_W$.
2. α_2 : What if the rates for men and women were switched?
 $x'_M = x_W$ and $x'_W = x_M$.
3. α_3 : What if the gender gap in the rate was different?
 $x'_M = x_M + k$ and $x'_W = x_W - k$.
4. α_4 : What if the overall rate was different (the same gender gap)?
 $x'_M = x_M + k$ and $x'_W = x_W + k$.

An important step of our process is figuring out *how* to model a low-level operation with a specific data symmetry. Although a systematic treatment of how to express low-level tasks as data symmetries is beyond the scope of this paper, we offer the following as general guidance. Starting from the description of the input data, we consider the simplest features of interest. In this case, we want to investigate the gender gap and total employment, in addition to specific rates. From that, we find operations that change those features while leaving the rest of the dataset intact, to isolate specific effects in the visualization. We incrementally grow the set of data symmetries until we are satisfied. Finally, we note that an incremental design based on reducing the confuser set would naturally expose the new “visible” data symmetries (those no longer belonging to a confuser) as a natural candidate for interpretation as a low-level operation.

While the redesign strategy here works from data symmetries α to visual symmetries ω to find jumblers, one can also imagine working the other way, finding misleaders by checking if the clearest visual symmetries ω correspond to the most important data symmetries α .

5.2 Design iterations

5.2.1 Design #1: bars and dots

Shown left-most in Fig. 7, this design matches data symmetry 1 with changes in distance between the dot and the axis (for changing x_M) or changes of length of the bar (for changing x_W). Distance and length

are not the same, but both rank highly with Cleveland and McGill, so we consider these ω s to respect our Correspondence Principle. Swapping male and female employment rates with data symmetry 2 corresponds to swapping impressions of length with impressions of distance from a common axis, which in our case means that a dot will be placed *inside* the bar. Some means of maintaining the visual contrast of dots inside bars (as compared to the dots against the white background) may be needed for legibility, but this does not itself violate Correspondence. However, changing the gender gap with data symmetry 3 is not directly matched with any element of the visual encoding, except perhaps the gap between the dot and the top of the bar, via some Gestalt consideration of negative space. This suggests a Correspondence failure: the design does not support visually investigating the *gap* itself. Data symmetry 4 reveals a similar Correspondence failure, as changes in the overall rate of management employment are at best matched to the midpoint between the dot and top of the bar, for which the visualization offers no direct support.

5.2.2 Design #2: two sets of bars, different scales

Side-by-side bar charts, one for each gender, replace the overlaid bar and dot plots. By using the same encoding for each gender, this design displays for data symmetry 2 (swapping x_M and x_W) a straightforward ω : swap the two bars. Data symmetries 3 and 4 remain unresolved.

5.2.3 Design #3: dots with line segments, same scale

This replaces bars by dots, improving the visual symmetry matched to data symmetries 1 and 2. We argue that with respect to these particular data symmetries, this design respects Correspondence in the best way known: distance from a common scale. The simple addition of a gray line drawn between the two dots dramatically remedies the previous problems with data symmetry 3: changing the gender gap changes the length of the line, *in addition* to moving the two dots. Data symmetry 4 corresponds to moving an entire dot-line-dot assembly left or right, which is not a simple visual affordance.

5.2.4 Design #4: Plain Scatterplot

Failures of Correspondence were identified for particular data symmetries in previous designs, but they all fail the Invariance Principle: the visual impression of the data is changed by permuting the country order. The scatterplot, however, is by nature *invariant* to list ordering, so it respects Invariance with respect to permutation hallucinators. Note that Designs #3 and #4 seem comparable in terms of Tufte’s notion of *chartjunk*. In other words, the presence or absence of a hallucinator is unrelated to the amount of unnecessary decoration on the plot. We agree that removing chartjunk generally improves visualizations, but solving Invariance failures involves the structure of a visualization rather than its embellishments. The scatterplot also offers good visual symmetries for data symmetry 1. Changing x_M or x_W moves a dot horizontally

or vertically; in either case, the position along a common scale is the highest-ranking affordance from Cleveland and McGill. The other data symmetries show weakness with respect to Correspondence, however, because they correspond to moving a dot along either line $x_W = -x_M$ or $x_W = x_M$, for which the design offers no clear visual help.

5.2.5 Design #5: Decorated Scatterplot

Data symmetries 2, 3 and 4 correspond to changing coordinates along a new set of axes, namely $x_W = x_M$ and $x_W = -x_M$. Our final design creates explicit visual symmetries for these data symmetries by adding support lines along those coordinates. Cleveland and McGill predict that adding such support lines facilitates interpretation. Indeed, this design exposes the relationships, hidden by any single ranking, that permit informed comparisons between countries. USA and GBR differ in their gender gap, while USA and ITA differ in the total rate of employment in management. Our design does not include a second perpendicular set of support lines to the $x_W = -x_M$ axis, but a meaningful visual symmetry still encodes data symmetry 4: the position of the *intersection* between the per-country support lines and the $x_W = x_M$ line. Though perhaps more subtle than the other visual symmetries, we consider this use of position as satisfying Correspondence.

This example illustrates how our algebraic process helps decompose the visualization evaluation and design challenge into smaller problems of reconciling specific data symmetries (low-level tasks) with visual symmetries (visual affordances), in a manner that requires less inspiration and more mechanistic attention to detail. On the other hand, we do not claim that the final design is superior to all possible alternatives: our improvements are relative to the choice of low-level tasks (and hence of data symmetries). Requiring different data symmetries, or prioritizing them differently, can lead to different designs. For example, one data symmetry not matched by any design above is changing the *relative gap size* $(x_M - x_W)/(x_M + x_W)$.

6 DISCUSSION AND LIMITATIONS

This section relates our principles to similar ideas in the literature, and discusses design decisions that strictly speaking lie outside of our principles, but which nevertheless affect the algebraic design process. We also discuss situations where the principles are too limiting or not powerful enough to explain or guide current designs.

6.1 Distinctions and tradeoffs in applying the theory

Applying our design principles relies on distinguishing the data D from the representation R , and understanding the differences between mappings on different spaces. Carefully making these specifications for a given design challenge is in a sense outside our principles: these decisions define the setting in which the principles operate. In the same way that α and ω model tasks and visual affordances, the description of D becomes a model for the problem domain. The value of thoughtfully characterizing these notions has been argued before with different terminology, most notably by Munzner [33].

Consider visualizing temperatures across the globe with a colormap. The colormap legend shows the colors ranging from the minimum to the maximum temperature, but suppose there is no numeric scale. The visualization is unchanged by multiplying the temperatures by a positive non-zero constant, or adding an offset. We could say the visualization fails Unambiguity, with the confusers just described. Or, we could interpret the given temperature values as numbers on some arbitrarily chosen scale (degrees Fahrenheit or Celsius). Then the visualization satisfies the Invariance Principle with respect the arbitrariness of the temperature scale, rather than failing Unambiguity. The difference lies in how the given numeric values should be understood. Considering representations of a set, we used sensitivity with respect to reordering to exemplify Invariance failure (Fig. 1(a)), but for methods like the LineUp system of Gratzl et al. for studying rankings [14], showing the ordering is the goal of the visualization rather than an Unambiguity failure.

By encouraging a designer to explicitly consider data symmetries, representation mappings and visual symmetries, we hope to understand and inform how designers choose algorithmic elements, or innovate new ones, when creating visualization methods. For example, faced

with a need to blend colors and opacities, a default choice might be the ubiquitous “over” operator [37]. Seeing that in scatterplots its non-commutativity causes Invariance failures (Fig. 2(a) top), our solution (Fig. 2(a) bottom) accumulates color separately from opacity, then combines the average opacity and color in a separate pass. The commutativity of addition erases ordering information from the visualization. For their Splatterplots method, Mayorga and Gleicher render overplotting results with a commutative color blending in perceptual color space [30]. Zheng et al., on the other hand, develop an order-enhancing variant of the over operator for volume rendering [62] to remove a specific confuser: shuffling voxels along the viewing ray. Our theory reconciles the contrasting choices of blending operations in the different domains, and describes them in a uniform way.

The issue of data features and intentional confusers is raised by the Line Integral Convolution (LIC) flow visualization in Fig. 3(c). We removed a confuser by modifying LIC to indicate flow velocity, but this creates a hallucinator for flow field topologists, for whom velocity is a representation detail, and the interesting data symmetries α include changing the location and type of critical points (where the velocity is zero). Design goals in the general area of “feature-based visualization” are sometimes more concisely defined in terms of their intentional confusers – the data transformations that should leave unchanged the features of interest – than in terms of the α that might influence the resulting visualization. In this setting, one could consider extracting two different kinds of features from a single flow field. A strict application of our theory might then say there are actually two different underlying “data”, both represented by the same field, which could be confusing.

Even with clear distinctions between data and representation, the aims of our principles may require tradeoffs. Unambiguity of the parallel coordinate plot (Fig. 3(b)) was achieved by adding colors to the individual lines, at the cost of some Invariance. Similarly, while the Correspondence Principle describes how abstract tasks on the data α should be shown with visual changes ω , there may be many equivalently effective (α, ω) pairs. Our model cannot by itself resolve such tensions, but we suggest that addressing them should be an explicit rather than an incidental part of visualization design. The tunable abstraction of Böttger et al. [4] is especially interesting in this light, in which the user can select different visual symmetries, morphing between a topology-centric and a geography-centric map layout. This lets the user “choose their confuser”, a workable compromise when all desired data symmetries cannot be simultaneously satisfied.

6.2 Generalizations of the design equation (1)

As defined, (1) is a strict equality, so any small change in a visualization is sufficient to trigger a failure of Invariance, or adherence to Unambiguity. This is impractical, so we explore here some alternatives that hinge on relaxing the notion of equality between two visualizations.

Consider using our model to design visualizations for users with color-deficient vision. In (1), we can include composition with the projection p to the color-deficient perceptual color space, giving

$$p \circ v \circ r_2 \circ \alpha = \omega \circ p \circ v \circ r_1. \quad (2)$$

We implicitly used a similar idea in describing ellipsoid glyphs and their Unambiguity failure (Fig. 1(b)). The ellipsoid glyph renderings are exactly equal when projected onto the subspace of outlines and shadows (one way of modeling bas-relief ambiguity [1]). As our models of perception become more detailed, more sophisticated notions of equivalence may be used. Using computational models of change blindness, for example, may help designers avoid ambiguous visualizations [27].

Adding a projection step is a sensible choice whenever justified by mathematical models of perception. Still, many interesting properties of visualizations are not so easily captured by mathematical models. In that case, user studies may provide evidence of distinguishability (or invariance). Recently, Wickham, Hofmann and co-authors have started investigating user studies as direct methods for collecting statistical evidence [60, 18]. Combining our model with the lineup protocol, for example, could give a direct way for testing perceptual distinguishability of data transformation symmetries α . If the original untransformed

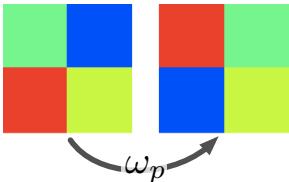


Fig. 8: Hues have no intrinsic order, so the rainbow map on ordered discrete data causes permutation misleaders ω_p (left). Smooth functions on connected domains have no permutation misleaders (right).

data cannot be identified when hiding amidst the transformed versions, then the data symmetries are effectively confusers of the visualization.

Our model can also leverage previous work on preattentiveness [16]. By explicitly matching basic data symmetries to preattentive visual stimuli (and other data symmetries to stimuli that are not), a designer can create a visualization that, though static, obeys the mantra of “overview first, details on demand”. We can also use a weaker notion of equality, as suggested above, by defining equality of visualizations by “cannot be distinguished in a small amount of time”. Preattentive visual channels are the most likely to produce visualizations that satisfy Unambiguity. A good example of this approach is Holten and van Wijk’s study of different depictions of directed edges [19].

Our model gives a decomposition of user interaction with a visualization in terms of (α, ω) pairs. Because these decompositions can also be used to design visualizations that are almost equal to one another, the model can also suggest user studies to compare the efficacy of different visual symmetries, controlling for the other visual symmetries existing in the visualization. These examples highlight how a qualitative, mathematical description of visualization design is not meant to replace experimental studies, but rather to complement them, and perhaps to provide a bridge between the majority of the visualization papers that do not provide quantitative evaluations and the few that do [22].

6.3 Generalizations of Representation Invariance

The presence in (1) of two different representation functions r_1 and r_2 models how visualization designers generally cannot control which particular representation they are given. We now explore some ways in which the same idea may apply more generally.

Uncertainty visualization and representation is an increasingly active area of research [38]. If we model uncertainty with a probability distribution and consider datasets as noisy samples from a population, then the goal of uncertainty visualization could be understood as *invariance under different samplings*. In other words, if we model the population of interest as the input data, and the sampling process as a “representation mapping”, then visualizations with no depiction of uncertainty will fail to indicate the possible range of representative samplings, and thus risk Invariance failure. Bandwidth selection in kernel density estimation (Fig. 2(b)) provides a specific example of this phenomenon, in which choosing the bandwidth is a tradeoff between too narrow (Invariance failure) and too wide (Unambiguity failure).

Similarly, many visualization techniques depend on parameters for computation or random seeds for initialization. If we consider these processes as analogous to representation mappings, then Invariance can generalize the basic idea that visualizations should be robust to changes in parameters and initialization. Finally, concepts which operate at a higher level than visual stimuli offer a range of interesting possibilities for resolving Invariance. An interesting recent example of this is how Lin et al. select semantically-resonant colors to remove the arbitrariness of mapping categories to colors [25].

6.4 Reconsidering the rainbow colormap

Despite warnings [3], the rainbow colormap remains ubiquitous. Our theory offers a possible reason (Fig. 8). While it is true that discretely sampled hues (e.g. red, green, yellow, blue) do not enjoy a natural perceptual ordering, the rainbow colormap in scientific visualization is typically used for smooth functions over connected domains.

In the smooth setting, simple permutation symmetries of the data are impossible: one cannot change a portion of a smooth dataset without changing its neighborhood. Over smooth domains, the application of

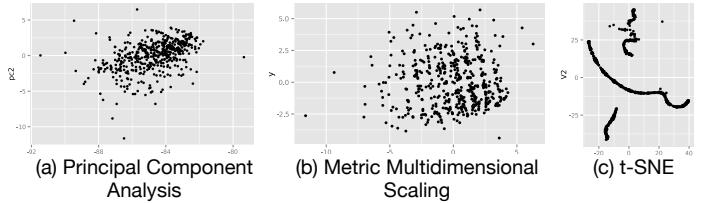


Fig. 9: Three evidently different 2D projections of a 500-point sample of the diamond dataset [59]: the first two principal components (PCA) (a), Metric MDS (MMDS) (b), and t -distributed stochastic neighbor embedding (t-SNE) [53] (c). We plan to apply our model to characterize exactly *how* these methods differ.

a colormap necessarily creates local copies of the colormap legend, which disambiguate ordering. In terms of our model, the matching (α, ω) possible on smooth spaces are more restricted than those on discrete spaces. We are not excusing the other perceptual flaws of rainbow colormaps (such as their uncontrolled luminance variation), but note that their continued use in scientific visualization may be associated with the underlying smooth data domains.

7 FUTURE WORK

We have seen here how our algebraic model explains many designs in current visualization research and practice. Longer term, our model can be a launching point for new investigations and applications. We would like to reconcile our theory with previous ones, both qualitative [28, 52, 33] and quantitative [8], in a comprehensive way by characterizing exactly when and how the theories overlap in their descriptive power or predictions. Understanding our model’s interface with the nascent information theory of visualization [9] may be valuable, for either algebraically constraining the large set of ω that conserve entropy but decrease legibility, or for quantifying the information in the (α, ω) pairs that otherwise seem comparable according to our principles.

Our approach may also help answer specific visualization questions. For example, we hope to study concrete differences between dimensionality reduction techniques (Fig. 9), by extending our model to infinitesimal α of data values and infinitesimal ω of point positions, to compute partial derivatives of the mappings, which may provide a uniform way of describing what the mappings effectively minimize. Informative precedent may be found in previous energy-minimizing graph drawing methods that visually reveal graph clusters [34].

Because (α, ω) pairs provide explicit, unambiguous descriptions, our algebraic process could enrich visualization design pedagogy. When evaluating visualizations, their relative advantages, limitations, and tradeoffs would be clearer, so class discussions and project evaluations can be more productive. We also intend to revisit design studies with an eye towards the associated (α, ω) pairs: the template of Section 5 can provide richer critiques through the lens of data symmetries. Conversely, past and future perception research in visualization can be evaluated through the lens of visual symmetries. Finally, our work so far does not address interaction, which is an essential part of applied data visualization. We expect that expanding our theory to model interactions will be challenging but rewarding.

ACKNOWLEDGMENTS

We thank the reviewers for their constructive feedback. Conversations with Tamara Munzner, Stephen Ingram, Hadley Wickham, Çağatay Demiralp, Xavier Tricoche and Thomas Schultz have helped shape this work. Discussion at the 2009 Dagstuhl Scientific Visualization Seminar 09251 focused our mathematical thinking. Kevin J. Lynch of Keming Labs brought our attention to Jonathan Schwabish’s design study.

More examples and details of algebraic visualization design may be found at <http://algebraicvis.net>.

REFERENCES

- [1] P. N. Belhumeur, D. J. Kriegman, and A. L. Yuille. The bas-relief ambiguity. In *Proc. Comp. Vis. and Patt. Rec.*, pages 1060–1066, 1997.
- [2] J. Bertin. *Sémiologie Graphique. Les diagrammes, les réseaux, les cartes.* Gauthier-Villars, 1967.

- [3] D. Borland and R. M. Taylor, II. Rainbow color map (still) considered harmful. *IEEE Comp. Graph. and App.*, 27(2):14–17, 2007.
- [4] J. Böttger, U. Brandes, O. Deussen, and H. Ziegold. Map warping for the annotation of metro maps. *IEEE Comp. Graph. and App.*, 28(5):56–65, 2008.
- [5] M. Brehmer and T. Munzner. A multi-level typology of abstract visualization tasks. *IEEE T. Vis. Comp. Graph.*, 19(12):2376–2385, 2013.
- [6] B. Cabral and L. C. Leedom. Imaging vector fields using line integral convolution. In *Proc. ACM SIGGRAPH*, pages 263–270, 1993.
- [7] Çağatay Demiralp, J. F. Hughes, and D. H. Laidlaw. Coloring 3D line fields using Boy’s real projective plane immersion. *IEEE T. Vis. Comp. Graph.*, 15(6):1457–1464, 2009.
- [8] Çağatay Demiralp, C. E. Scheidegger, G. L. Kindlmann, D. H. Laidlaw, and J. Heer. Visual embedding: A model for visualization. *IEEE Comp. Graph. and App.*, 34(1):10–15, 2014.
- [9] M. Chen and H. Jänicke. An information-theoretic framework for visualization. *IEEE T. Vis. Comp. Graph.*, 16(6):1206–1215, 2010.
- [10] E. Cheng. Categories. In T. Gowers, J. Barrow-Green, and I. Leader, editors, *The Princeton companion to mathematics*. Princeton University Press, 2010.
- [11] W. S. Cleveland and R. McGill. Graphical perception: Theory, experimentation, and application to the development of graphical methods. *J. American Statistical Association*, 79(387):531–554, 1984.
- [12] J. J. Gibson. *The Ecological Approach To Visual Perception*, chapter 8: The Theory of Affordances. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1986.
- [13] M. Graham and J. Kennedy. Using curves to enhance parallel coordinate visualisations. In *Proc. Info. Vis.*, pages 10–16. IEEE, 2003.
- [14] S. Gratzl, A. Lex, N. Gehlenborg, H. Pfister, and M. Streit. Lineup: Visual analysis of multi-attribute rankings. *IEEE T. Vis. Comp. Graph.*, 19(12):2277–2286, 2013.
- [15] C. G. Healey. Choosing effective colours for data visualization. In *Proc. IEEE Visualization*, pages 263–270, 1996.
- [16] C. G. Healey, K. S. Booth, and J. T. Enns. High-speed visual estimation using preattentive processing. *ACM T. Comp.-Hum. Int.*, 3(2):107–135, 1996.
- [17] J. Helman and L. Hesslink. Representation and display of vector field topology in fluid flow data sets. *Computer*, 22(8):27–36, 1989.
- [18] H. Hofmann, L. Follett, M. Majumder, and D. Cook. Graphical tests for power comparison of competing designs. *IEEE T. Vis. Comp. Graph.*, 18(12):2441–2448, 2012.
- [19] D. Holten and J. J. van Wijk. A user study on visualizing directed edges in graphs. In *Proc. SIGCHI Conference on Human Factors in Computing Systems*, pages 2299–2308. ACM, 2009.
- [20] B. K. P. Horn. *The Psychology of Computer Vision*, chapter 4: Obtaining Shape from Shading Information, pages 115–155. McGraw-Hill, 1975.
- [21] J. Hullman, S. Drucker, N. Henry Riche, B. Lee, D. Fisher, and E. Adar. A deeper understanding of sequence in narrative visualization. *IEEE T. Vis. Comp. Graph.*, 19(12):2406–2415, 2013.
- [22] T. Isenberg, P. Isenberg, J. Chen, M. Sedlmair, and T. Möller. A systematic review on the practice of evaluating visualization. *IEEE T. Vis. Comp. Graph.*, 19(12):2818–2827, 2013.
- [23] G. Kindlmann. Superquadric tensor glyphs. In *Proc. EG/IEEE TCVG Symposium on Visualization*, pages 147–154, 2004.
- [24] M.-H. Kiu and D. C. Banks. Multi-frequency noise for LIC. In *Proc. IEEE Visualization*, pages 121–126, 1996.
- [25] S. Lin, J. Fortuna, C. Kulkarni, M. Stone, and J. Heer. Selecting semantically-resonant colors for data visualization. *Comp. Graph. Forum*, 32(3):401–410, 2013.
- [26] Z. Liu, N. J. Nersessian, and J. T. Stasko. Distributed cognition as a theoretical framework for information visualization. *IEEE T. Vis. Comp. Graph.*, 14(6):1173–1180, 2008.
- [27] L.-Q. Ma, K. Xu, T.-T. Wong, B.-Y. Jiang, and S.-M. Hu. Change blindness images. *IEEE T. Vis. Comp. Graph.*, 19(11):1808–1819, 2013.
- [28] J. Mackinlay. Automating the design of graphical presentations of relational information. *ACM T. Graph.*, 5(2):110–141, 1986.
- [29] J. Mackinlay, P. Hanrahan, and C. Stolte. Show me: Automatic presentation for visual analysis. *IEEE T. Vis. Comp. Graph.*, 13(6):1137–1144, 2007.
- [30] A. Mayorga and M. Gleicher. Splatterplots: Overcoming overdraw in scatter plots. *IEEE T. Vis. Comp. Graph.*, 19(9):1526–1538, 2013.
- [31] M. Meyer, M. Sedlmair, and T. Munzner. The four-level nested model revisited: blocks and guidelines. In *Proc. 2012 BELIV Workshop*, pages 11:1–11:6, 2012.
- [32] T. Möller, R. Machiraju, K. Mueller, and R. Yagel. Evaluation and design of filters using a Taylor series expansion. *IEEE T. Vis. Comp. Graph.*, 3(2):184–199, 1997.
- [33] T. Munzner. A nested model for visualization design and validation. *IEEE T. Vis. Comp. Graph.*, 15(6):921–928, 2009.
- [34] A. Noack. An energy model for visual graph clustering. In *Graph Drawing*, pages 425–436. Springer, 2004.
- [35] E. Parzen. On estimation of a probability density function and mode. *Annals of mathematical statistics*, 33(3):1065–1076, 1962.
- [36] C. Pierpaoli. Oh no! One more method for color mapping of fiber tract direction using diffusion MRI data. In *Proc. 5th ISMRM*, page 1741, 1997.
- [37] T. Porter and T. Duff. Compositing digital images. In *Proc. ACM SIGGRAPH*, pages 253–259, 1984.
- [38] K. Potter, P. Rosen, and C. R. Johnson. From quantification to visualization: A taxonomy of uncertainty visualization approaches. In *Uncertainty Quantification in Scientific Computing*, pages 226–249. Springer, 2012.
- [39] C. Rampell. Comparing the world’s glass ceilings. <http://economix.blogs.nytimes.com/2013/04/02/comparing-the-worlds-glass-ceilings>, 2013.
- [40] T. Schultz and G. L. Kindlmann. Superquadric glyphs for symmetric second-order tensors. *IEEE T. Vis. Comp. Graph.*, 16(6):1595–1604, 2010.
- [41] J. Schwabish. Mind the gap: An economic chart remake. <http://thewhyaxis.info/gap-remake>, 2013.
- [42] R. N. Shepard and L. A. Cooper. *Mental Images and Their Transformations*. MIT Press, 1982.
- [43] B. Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM T. Graph.*, 11(1):92–99, 1992.
- [44] B. Shneiderman. The eyes have it: a task by data type taxonomy for information visualizations. In *Proc. IEEE Symposium on Visual Languages*, pages 336–343, 1996.
- [45] S. S. Stevens. On the theory of scales of measurement. *Science*, 103(2684):677–680, 1946.
- [46] M. Tory and T. Möller. Rethinking visualization: A high-level taxonomy. In *Proc. Info. Vis.*, pages 151–158, 2004.
- [47] B. E. Trumbo. A theory for coloring bivariate statistical maps. *The American Statistician*, 35(4):220–226, 1981.
- [48] E. R. Tufte. *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, CT, 2nd edition, 2001.
- [49] G. Turk and D. Banks. Image-guided streamline placement. In *Proc. ACM SIGGRAPH*, pages 453–460, 1996.
- [50] A. Tversky. Features of similarity. *Psychological review*, 84(4):327, 1977.
- [51] B. Tversky, J. E. Corter, L. Yu, D. L. Mason, and J. V. Nickerson. Representing category and continuum: Visualizing thought. In *Diagrammatic Representation and Inference*, volume 7352 of *Lecture Notes in Computer Science*, pages 23–34. Springer, 2012.
- [52] B. Tversky, J. B. Morrison, and M. Betrancourt. Animation: can it facilitate? *Intl. J. Hum.-Comp. Stud.*, 57(4):247–262, 2002.
- [53] L. van der Maaten and G. Hinton. Visualizing data using t-SNE. *J. Mach. Learn. Res.*, 9(Nov):2579–2605, 2008.
- [54] J. J. van Wijk and H. van de Wetering. Cushion treemaps: Visualization of hierarchical information. In *Proc. Info. Vis.*, pages 73–78, 1999.
- [55] C. Ware. Toward a perceptual theory of flow visualization. *IEEE Comp. Graph. and App.*, 28(2):6–11, 2008.
- [56] C. Ware. *Information visualization: perception for design*. Elsevier, 2012.
- [57] M. Wattenberg and D. Fisher. A model of multi-scale perceptual organization in information graphics. In *Proc. Info. Vis.*, pages 23–30, 2003.
- [58] E. W. Weisstein. Cayley cubic. Wolfram Mathworld. <http://mathworld.wolfram.com/CayleyCubic.html> (accessed 24 July 2014).
- [59] H. Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer New York, 2009.
- [60] H. Wickham, D. Cook, H. Hofmann, and A. Buja. Graphical inference for infovis. *IEEE T. Vis. Comp. Graph.*, 16(6):973–979, 2010.
- [61] G. Wyszecki and W. S. Stiles. *Color science: Concepts and methods, quantitative data and formulae*. Wiley New York, 1982.
- [62] L. Zheng, Y. Wu, and K.-L. Ma. Perceptually-based depth-ordering enhancement for direct volume rendering. *IEEE T. Vis. Comp. Graph.*, 19(3):446–459, 2013.
- [63] C. Ziemkiewicz and R. Kosara. Embedding information visualization within visual representation. In Z. W. Ras and W. Ribarsky, editors, *Advances in Information and Intelligent Systems*, volume 251 of *Studies in Computational Intelligence*, pages 307–326. Springer, 2009.