Visualization of Equivalence in 2D Bivariate Fields

Boyan Zheng[1] ID, Bastian Rieck[2] ID, Heike Leitte[2] ID, Filip Sadlo[1] ID

[1] Heidelberg University, Germany
[2] TU Kaiserslautern, Germany

Abstract
In this paper, we show how the equivalence property leads to the novel concept of equivalent regions in mappings from $\mathbb{R}^n$ to $\mathbb{R}^n$. We present a technique for obtaining these regions both in the domain and the codomain of such a mapping, and determine their correspondence. This enables effective investigation of variation equivalence within mappings, and between mappings in terms of comparative visualization. We implement our approach for $n = 2$, and demonstrate its utility using different examples.

CCS Concepts
• *Human-centered computing* → *Visualization techniques;* • *Applied computing* → *Mathematics and statistics;*

1. Introduction

One of the most basic problems in data analysis is to find a given value in a dataset. If the value is a scalar and the dataset a scalar field, the result are contours. That is, except for the values of local extrema, there are manifolds consisting of infinitely many points exhibiting such a value. This, on the other hand, means that in general, such analysis requires exploration by means of, e.g., interactive isocontour visualization. If, in contrast, the given value is a vector of dimension n, and the dataset is an n-variate (n-vector) field in n-dimensional space, the loci exhibiting that value are, in general, isolated points. This can be explained by regarding the n-variate field as a set of n scalar fields in n-dimensional space, and the given vector as the intersection of respective isocontours. If the contours intersect at more than one location, the respective points have the same vector value, i.e., are equivalent, and we denote their count as the multiplicity of the mapping for this given vector value.

The properties central to this paper are this equivalence and multiplicity in nD n-variate mappings, or in other words, nD vector fields. For simplicity, we denote these fields as vector fields where advantageous, although the main application of this work are 2D bivariate mappings, such as temperature and pressure defined on a 2D domain, i.e., $n = 2$. Our approach exploits the "dual" representation of such mappings, where position is interpreted as vector value, and vector value as position. This dual representation is closely related to continuous scatterplots [BW08]. We complement the discontinuities in continuous scatterplots due to Lehmann and Theisel [LT10] with additional curves of discontinuity, which enables us to confine the regions of uniform multiplicity both in the codomain and the domain. The resulting equivalence relation between these regions finally establishes our *equivalent regions*, whose utility for investigating equivalence within and across such mappings we demonstrate with different examples.

The contributions of this paper include:
- the concept of equivalent regions in nD n-variate mappings,
- extraction of equivalent regions, both in domain and codomain,
- interactive exploration based on the mapping between these, and
- application of equivalent regions for comparative visualization.

2. Related Work

This paper was inspired by the observations that led to the development of continuous scatterplots [BW08] and continuous parallel coordinates [HW09]. The authors of both publications analyze mappings of the form $\mathbf{x} \colon \mathbb{R}^n \to \mathbb{R}^m$, with $n \neq m$ typically, in a continuous manner instead of a discrete one by taking into account the structure of the domain. In particular, continuous scatterplots focus on a precise description of density variations arising from $\mathbf{x}$, i.e., from the mapping. Lehmann and Theisel [LT10], targeting $n = m = 2$ and $n = 2, m = 3$, were the first to describe how the mapping induces variations in continuity. Their work is thus a precursor to our paper; we extend their concepts in terms of the implications of equivalence within mappings, which permits us to describe more phenomena that arise within this context (see Section 4 and in particular Section 4.7 for a more detailed discussion of differences).

Recently, the study of multi-field datasets, i.e., mappings in which $m \geq 2$, has been improved by the introduction of Reeb spaces [EHP08], a concept analogous to Reeb graphs [PSBM07] that are commonly used in the analysis of scalar fields. This led to the development of joint contour nets [DCK*12, CD13, CD14], a technique that quantizes all contours of a bivariate function individually, thereby permitting an analysis of simultaneous changes in both variables. An extension of this work was introduced by Carr et al. [CGT*15] in the form of fiber surfaces, which also focus on bivariate functions, i.e., $m = 2$, with a specific restriction to $n = 3$. These surfaces permit segmenting the range of the mapping into

regions of similar behavior, which is somewhat similar to the regions of equivalence that we extract for the case $n = m$. However, the authors note that the extraction of fiber surfaces requires using the marching cubes [LC87] algorithm and does not always result in an *exact* solution, whereas our method is guaranteed to result in geometrically exact regions. While the missing exactness of fiber surfaces was solved in a followup paper by Tierny et al. [TC17], their use case ($n = 3$, $m = 2$) is still different from ours.

3. Motivation

Let us start with a simple 1D example to motivate our approach and build intuition of its applicability and interpretation. It consists of a 1D scalar-valued function, i.e., mapping a scalar in the domain Ω to a scalar in the codomain Ψ. Figure 1a illustrates the graph of such a function, in traditional domain-based representation, while Figure 1b illustrates it in the codomain. In Figure 1a, we identify what we later define to be boundary points $\mathcal{B}_1$ and $\mathcal{B}_2$, and silhouette points $\mathcal{S}_1$ and $\mathcal{S}_2$. Whereas the value of the boundary point $\mathcal{B}_2$ is unique, and therefore does not provide projections on the graph, the value of the boundary point $\mathcal{B}_1$ appears three times in the data, leading to two projected boundary points $\overline{\mathcal{B}_1^1}$ and $\overline{\mathcal{B}_1^2}$. Similarly, the value of the silhouette point $\mathcal{S}_2$ is unique, but the value of $\mathcal{S}_1$ appears twice, leading to one projected silhouette point $\overline{\mathcal{S}_1}$.

Figure 1b illustrates the "folded domain manifold", i.e., that the function attains the intervals α, β, and γ in the codomain one, two, or three times (circled numbers). These values represent what we later define to be the multiplicity field in the codomain. Mapping these values to the domain leads to respective intervals a, b, c, and d in the domain (Figure 1a), with multiplicity indicated by circled numbers. Notice that in the domain, region b exhibits a symmetry point ($\mathcal{S}_2$) and thus "folds on itself", leading to multiplicity 2. Region a also exhibits a symmetry point ($\mathcal{S}_1$), but at the same time "folds" on region c, leading to overall multiplicity 3. Finally, region d has multiplicity 1, i.e., it exhibits values that are unique.

From the mapping, we can determine correspondences. In our example, α corresponds to d, β corresponds to a and c, and γ corresponds to b. As a consequence, we will define the regions a and c to be equivalent, because they exhibit the same values, in a continuous manner. Equivalent regions provide the overall structure of equivalence in n-variate nD fields. In vector field topology [HH89], for example, separatrices split the domain into regions of qualitatively similar flow behavior. That is, instead of having to examine all possible streamlines within a vector field, it is, for many purposes, sufficient to investigate a single streamline for each of these regions, since all streamlines within such a region are similar. In our case, the equivalent regions provide the structure of equivalence. That is, if we are interested in a certain location in the domain, and this location is contained in a region with multiplicity μ, we know that there are μ points in total that exhibit that respective value. And the equivalent regions show the locations where these values are. At the same time, we know that if we move around in an equivalent region without leaving it, the number of points with identical value will not change, which strongly supports (interactive) analysis. For example, if we move around in region b, we know that there will be always one additional point having the same value and that this point will be on the opposite side of the symmetry point $\mathcal{S}_2$. Cross-
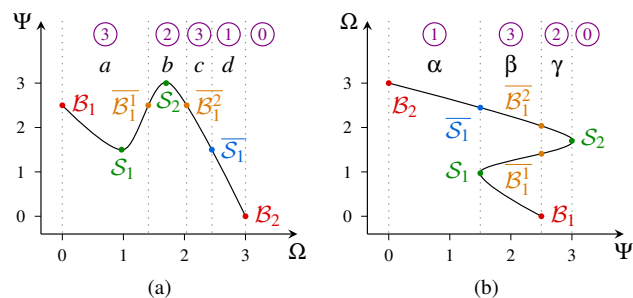

Figure 1: *Simple 1D example. (a) Function in domain Ω representation, with boundary points $\mathcal{B}_1$ and $\mathcal{B}_2$, silhouette points $\mathcal{S}_1$ and $\mathcal{S}_2$, projected boundaries $\overline{\mathcal{B}_1^1}$ and $\overline{\mathcal{B}_1^2}$, projected silhouette $\overline{\mathcal{S}_1}$, multiplicities (circled), and equivalent regions (letters). (b) Corresponding representation in codomain Ψ, illustrating "folding".*

ing the projected boundaries $\overline{\mathcal{B}_1^1}$ or $\overline{\mathcal{B}_1^2}$ will, instead, e.g., cause an additional point with identical value originate at a distinct location. For example, crossing $\overline{\mathcal{B}_1^2}$ from b to c will involve a point entering a at $\mathcal{B}_1$. We would like to refer the reader to the video in the supplemental material for further demonstrations of these properties.

Let us now assume that Figure 1a represents data from the stock market, i.e., the abscissa represents time, and the ordinate represents value. If we are interested in values that are attained often, we look for equivalent regions with high multiplicity. The equivalent regions in the domain then represent the time intervals at which these values occur. We can also find rare configurations by investigating regions with low multiplicity. One difficulty with equivalent regions, however, is that they are exact, and that large data with high variation will lead to many small equivalent regions (assume in our example, that the graph from Figure 1a is much longer with random variation). The reason for this is that each additional local minimum or maximum causes an additional "fold" and thus increases the multiplicity of at least one equivalent region and generates at least one new region. However, if one is interested in the fine-scale structure of such data, one has to take the many small regions into account, similar to vector field topology which exhibits extremely complex topological skeletons for, e.g., turbulent flow. Nevertheless, inspired by persistence concepts, a straightforward approach to obtain an overview with our technique, is to smooth the data, e.g., using Gaussian smoothing, to get rid of small "bumps", i.e., local minima and maxima that have low persistence. We exemplify the utility of such smoothing for large-scale analysis in Section 6.3, where we analyze climatic equivalent regions, i.e., climate zones.

4. Equivalence in nD n-variate Fields

The subject of our work are continuous n-dimensional mappings $\mathbf{u}(\mathbf{x})$, which assign to each position $\mathbf{x} \in \Omega \subset \mathbb{R}^n$ a vector value $\mathbf{u} \in \Psi \subset \mathbb{R}^n$, with Ω being the domain, and Ψ being the codomain, i.e., mappings $\mathbf{u}: \Omega \to \Psi$. Figure 2a shows an example of such a map.

4.1. Inverse Mapping and Multiplicity

From $\mathbf{u}(\mathbf{x})$, we can derive its inverse mapping $\mathbf{x}: \Psi \to \Omega$, which we denote $\mathbf{x}(\mathbf{u})$. Notice that, in general, this mapping maps a point in

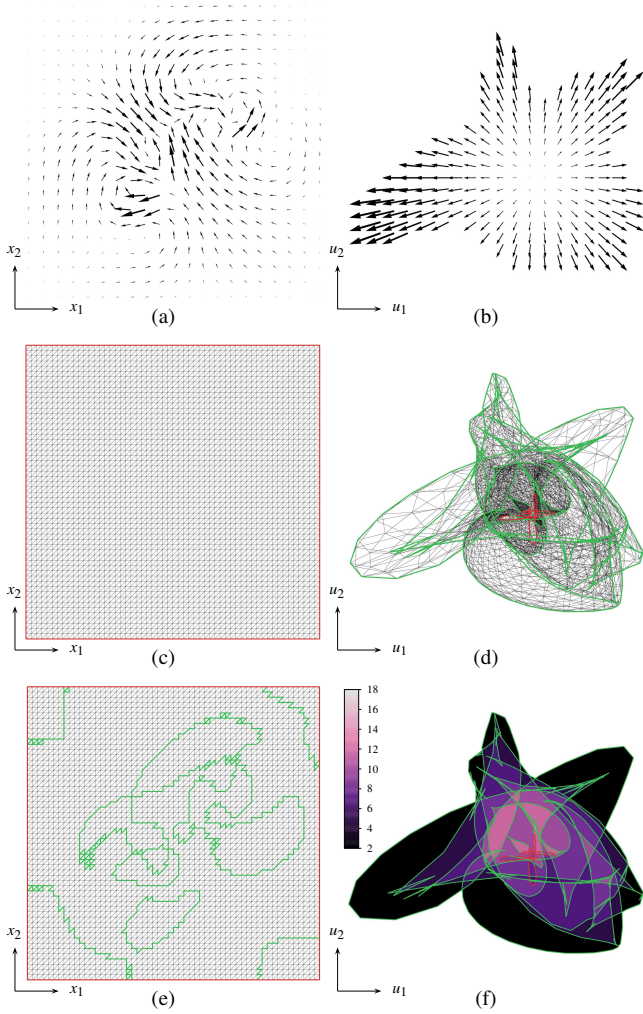**Figure 2:** *Mapping $\mathbf{u}(\mathbf{x})$ ($\mathbb{R}^2 \to \mathbb{R}^2$) in domain (left column) and codomain (right column). (a) Field in domain representation. Space by arrow position, value by arrow orientation/length. (b) Field in codomain representation, showing only value, due to varying cardinality of preimage. Mapping $\mathbf{u}(\mathbf{x})$ maps domain manifold (c) to codomain, i.e., distorts and "folds" it (d), leading to varying number of "layers", i.e., multiplicities $\mu(\mathbf{u})$ (f). As previously shown [LT10], silhouette curves (green) and boundary curves (red) ((d) and (e)) separate regions of uniform $\mu(\mathbf{u})$.*

Figure 3: *Mapping from Figure 2, same color-coding. (a) Multiplicity μ mapped to domain (from Figure 2f). Silhouette curves $\mathcal{S}$ and boundary curves $\mathcal{B}$ (Figure 2e) do not separate regions of uniform $\mu(\mathbf{x})$. (b) "Projection" of $\mathcal{S}$ (blue) and $\mathcal{B}$ (orange) on manifold "layers" (see also (d)). (c) Projected $\mathcal{S}$ and $\mathcal{B}$ from (b) separate regions of uniform $\mu(\mathbf{x})$ in domain. (f) Uniform regions extracted in codomain from $\mathcal{S}$ and $\mathcal{B}$ (Figure 2f), and uniform regions extracted in domain from projected $\mathcal{S}$, projected $\mathcal{B}$, and $\mathcal{B}$ (e). Equivalent region correspondence by colors ((e) and (f)).*

Ψ to a *set* of points in Ω, i.e.,

$$\mathbf{x}(\mathbf{u}) := \{\mathbf{x} \mid \mathbf{u}(\mathbf{x}) = \mathbf{u}\}. \tag{1}$$

Figure 2b shows the codomain of the field from Figure 2a. Notice that in such basic visualizations in the codomain, our arrows display $\mathbf{u}$ instead of $\mathbf{x}$, because for visualizing $\mathbf{x}$, one would need to draw a set of arrows at each point $\mathbf{u}$ in Ψ.

This leads us to the *multiplicity $\mu(\mathbf{u})$*, i.e., the cardinality of $\mathbf{x}(\mathbf{u})$:

$$\mu(\mathbf{u}) := |\{\mathbf{x} \mid \mathbf{u}(\mathbf{x}) = \mathbf{u}\}|. \tag{2}$$

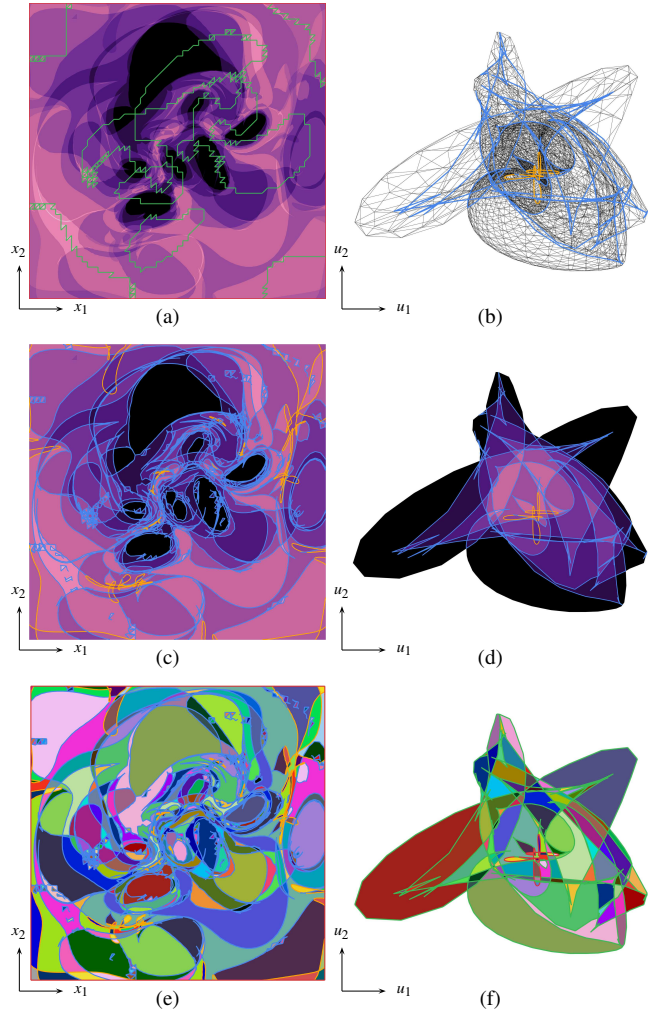Since $\mathbf{u}(\mathbf{x})$ maps each point in Ω to exactly one point in Ψ, and

since Ω is an *n*-manifold (Figure 2c), one can interpret this mapping as a "deformation". That is, in Ψ, this manifold is distorted, and at values $\mathbf{u}$ with $\mu(\mathbf{u}) > 1$, it is folded, leading to μ "layers" of the manifold (see Figure 2d and 2f).

4.2. Silhouette Curves and Boundary Curves

As has been shown by Lehmann and Theisel [LT10], the resulting regions with uniform $\mu(\mathbf{u})$ in the codomain (Figure 2f) are separated by *silhouette curves $\mathcal{S}$* and *boundary curves $\mathcal{B}$* (please notice that Lehmann and Theisel denote our silhouette curves as critical

curves). While the boundary curves are simply the boundary of Ω,

$$\mathcal{B}_{\Omega} := \partial \Omega \,, \tag{3}$$

silhouette curves $\mathcal{S}$ separate "front-facing" manifold regions from "back-facing" ones in the codomain (Figure 2d). If the mapping is given on a simplicial complex, silhouettes separate simplices with opposite handedness in Ψ. Generally, in the continuous case, and in grids with nonlinear interpolation, silhouette curves represent the zero-level contours of the Jacobian determinant [LT10]:

$$\mathcal{S}_{\Psi} := \{\mathbf{u} \mid \det \nabla \mathbf{u}(\mathbf{x}) = 0\} \,. \tag{4}$$

Whereas Lehmann and Theisel focused on continuous scatterplots, and thus on the codomain, we focus on the mapping between Ω and Ψ in terms of equivalence. Thus, whereas they were looking at $\mathcal{S}_{\Psi}$ and $\mathcal{B}_{\Psi}$, i.e., silhouette curves and boundary curves in the codomain (Figure 2f), our first step is to include $\mathcal{S}_{\Omega}$ and $\mathcal{B}_{\Omega}$, i.e., silhouette curves and boundary curves in the domain (Figure 2e):

$$\mathcal{S}_{\Omega} := \{\mathbf{x} \mid \det \nabla \mathbf{u}(\mathbf{x}) = 0\} \tag{5}$$

and

$$\mathcal{B}_{\Psi} := \{\mathbf{u} \mid \exists \mathbf{x} \in \mathcal{B}_{\Omega} : \mathbf{u} = \mathbf{u}(\mathbf{x})\} \,. \tag{6}$$

As a basis for the mapping between Ω and Ψ, we also obtain the multiplicity $\mu(\mathbf{x})$ in the domain:

$$\mu(\mathbf{x}) := |\{\boldsymbol{\xi} \mid \mathbf{u}(\boldsymbol{\xi}) = \mathbf{u}(\mathbf{x})\}| \,. \tag{7}$$

The color-coded values in Figure 3a show an example of multiplicity $\mu(\mathbf{x})$ in the domain. Interestingly, it can be seen that $\mathcal{S}_{\Omega}$ and $\mathcal{B}_{\Omega}$ (green and red in Figure 3a) do *not* separate regions of uniform $\mu(\mathbf{x})$ in the *domain*. Thus, we first need to derive respective curves that accomplish this. (Nevertheless, we will see later, that $\mathcal{S}_{\Omega}$ provide information about symmetries in equivalence.)

4.3. Projected Silhouettes and Projected Boundaries

Since the domain manifold is folded in Ψ (Figure 2d), we can "project" $\mathcal{S}_{\Psi}$ and $\mathcal{B}_{\Psi}$ on the manifold in the codomain, i.e., on all *other* "layers", providing projected silhouettes $\overline{\mathcal{S}}_{\Omega}$ in the domain:

$$\overline{\mathcal{S}}_{\Omega} := \{\mathbf{x} \mid \mathbf{u}(\mathbf{x}) \in \mathcal{S}_{\Psi}\} \setminus \mathcal{S}_{\Omega} \tag{8}$$

(blue in Figure 3c), and projected silhouettes $\overline{\mathcal{S}}_{\Psi}$ in the codomain:

$$\overline{\mathcal{S}}_{\Psi} := \{\mathbf{u} \mid \exists \mathbf{x} \in \overline{\mathcal{S}}_{\Omega} : \mathbf{u} = \mathbf{u}(\mathbf{x})\} \tag{9}$$

(blue in Figure 3b and 3d). Notice that "outlining" silhouettes are not contained in the projection (cf. Figures 2d and 3b). Correspondingly, we obtain projected boundaries $\overline{\mathcal{B}}_{\Omega}$ in the domain:

$$\overline{\mathcal{B}}_{\Omega} := \{\mathbf{x} \mid \mathbf{u}(\mathbf{x}) \in \mathcal{B}_{\Psi}\} \setminus \mathcal{B}_{\Omega} \,, \tag{10}$$

(orange in Figure 3c), and projected boundaries $\overline{\mathcal{B}}_{\Psi}$ in codomain:

$$\overline{\mathcal{B}}_{\Psi} := \{\mathbf{u} \mid \exists \mathbf{x} \in \overline{\mathcal{B}}_{\Omega} : \mathbf{u} = \mathbf{u}(\mathbf{x})\} \tag{11}$$

(orange in Figure 3b and 3d).

We already see in Figure 3c, that the union of $\overline{\mathcal{S}}_{\Omega}$ and $\overline{\mathcal{B}}_{\Omega}$ does the job of separating regions of uniform $\mu(\mathbf{x})$ in the domain. That is, we achieved the counterpart to Lehmann and Theisel's approach, i.e., we extended it from the codomain to the domain.

The explanation why the union of $\overline{\mathcal{S}}_{\Omega}$ and $\overline{\mathcal{B}}_{\Omega}$ is the required set of curves to delineate regions with uniform $\mu(\mathbf{x})$ in the domain is as follows (we assume nondegenerate configurations, i.e., $\mathcal{S}_{\Psi}$ and $\mathcal{B}_{\Psi}$ must not be (partially) congruent):

1. $\mathcal{S}_{\Omega}$ are contours (Equation 4), and thus closed curves (possibly including $\mathcal{B}_{\Omega}$). Assuming that $\mathbf{u}(\mathbf{x})$ is continuous and nondegenerate, it maps these curves to closed curves in the codomain.
2. As can be understood from Figure 2d, $\mathcal{S}_{\Omega}$ do not separate regions of different multiplicity $\mu(\mathbf{x})$ in the *domain* because neighboring points that are on either side of $\mathcal{S}_{\Omega}$ (Figure 2e) are located on the same side of the respective $\mathcal{S}_{\Psi}$ in the codomain, and therefore have the same multiplicity.
3. Removing (intersecting) closed curves ($\mathcal{S}_{\Omega}$ in Equation 8) from (intersecting) closed curves results in closed curves (since this cannot generate endpoints).
4. Therefore, and since silhouette curves and boundary curves separate different multiplicities in the codomain, the projections $\overline{\mathcal{S}}_{\Omega}$ and $\overline{\mathcal{B}}_{\Omega}$ have to separate regions of uniform $\mu(\mathbf{x})$ in the domain.

4.4. Equivalent Regions

The last steps to obtain the equivalent regions are to extract the regions of uniform μ in the domain as well as in the codomain, and to establish correspondence between them.

Since we now know that these regions are delineated by $\mathcal{G}_{\Omega} := \overline{\mathcal{S}}_{\Omega} \cup \overline{\mathcal{B}}_{\Omega} \cup \mathcal{B}_{\Omega}$ in the domain, and by $\mathcal{G}_{\Psi} := \mathcal{S}_{\Psi} \cup \mathcal{B}_{\Psi}$ in the codomain, the extraction problem reduces to finding all smallest closed loops in $\mathcal{G}_{\Omega}$ and in $\mathcal{G}_{\Psi}$. To this end, one can trace the resulting graphs $\mathcal{G}_{\cdot}$, taking always the "leftmost" branch, labeling the visited edges (each inner edge being visited twice), and obtain the regions from these loops. This results in a respective set of regions $\mathcal{R}_{\Omega}$ in the domain, and a set of regions $\mathcal{R}_{\Psi}$ in the codomain. Finally, correspondence $c_{\Omega \to \Psi}$ between a region $r_{\Omega} \in \mathcal{R}_{\Omega}$ and the respective region $r_{\Psi} \in \mathcal{R}_{\Psi}$ can be established by:

$$c_{\Omega \to \Psi}(r_{\Omega}) := \{r_{\Psi} \mid \exists \mathbf{x} \in r_{\Omega} : \mathbf{u}(\mathbf{x}) \in r_{\Psi}\} \,. \tag{12}$$

As discussed, $|c_{\Omega \to \Psi}(r_{\Omega})| = 1$, i.e., for each r_{Ω} there is exactly one corresponding r_{Ψ}. Conversely, the correspondence $c_{\Psi \to \Omega}$ between a region $r_{\Psi} \in \mathcal{R}_{\Psi}$ and region(s) $r_{\Omega} \in \mathcal{R}_{\Omega}$ is obtained by

$$c_{\Psi \to \Omega}(r_{\Psi}) := \{r_{\Omega} \mid \exists \mathbf{x} \in r_{\Omega} : \mathbf{u}(\mathbf{x}) \in r_{\Psi}\} \,, \tag{13}$$

and in this case, $|c_{\Psi \to \Omega}(r_{\Psi})| = \mu(\mathbf{u})$ with $\mathbf{u} \in r_{\Psi}$. Figure 3e and 3f shows an example of this mapping, correspondences by colors.

4.5. Equivalent Variation and Symmetry

So far, we obtained in the codomain regions r_{Ψ} with uniform multiplicity, as well as in the domain respective regions r_{Ω}, and established their correspondence in terms of $c_{\Omega \to \Psi}$ and $c_{\Psi \to \Omega}$. Let us now examine the interpretation of these regions and their interrelation. Figure 4a shows $\mathcal{R}_{\Omega}$ in the domain, and Figure 4b shows $\mathcal{R}_{\Psi}$ in the codomain, both with color-coded correspondences and respective arrow glyphs. Let us start with the green upper region in Figure 4a (and lower in 4b), denoted with (*i*). We marked the correspondence between the domain and the codomain for the points (1)–(5) of this region. We observe, that each of the points (2)–(4) is present two times in the domain, whereas it is present only once
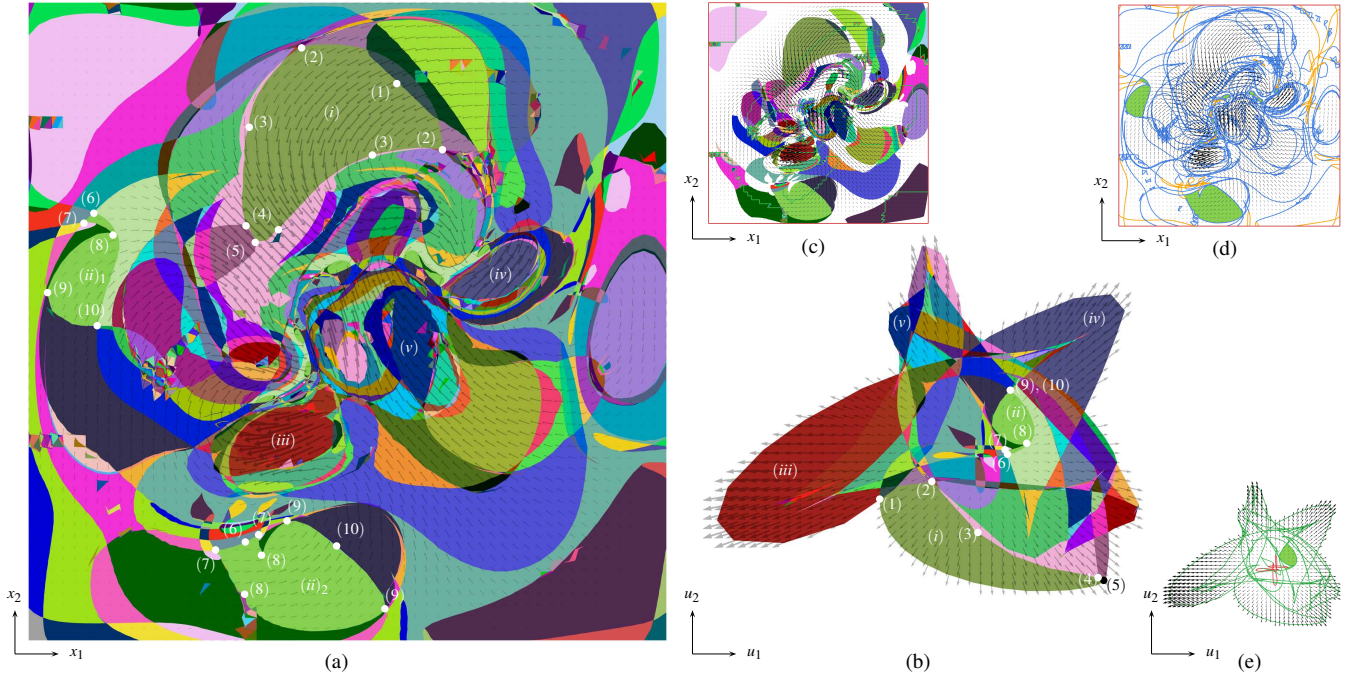
Figure 4: *Mapping from Figure 2. ((a) and (b)) Same as Figure 3e and 3f, but with arrow glyphs and point correspondences (1)–(10). ((d) and (e)) Equivalent regions (ii) from (a) and (b). (c) Equivalent regions that exhibit symmetry lines (green curves).*

in the codomain, i.e., the respective pairs are mapped to the same point in Ψ, or in other words, they exhibit the same vector value. This is consistent with the obtained $\mu(\mathbf{u}) = 2$ for this region. On the other hand, its points (1) and (5) are present only once both in the domain and the codomain. We conclude that region (i) folds on itself when mapped from Ω to Ψ (see also Figure 8c), and that there is a curve from (1) to (5) along which the region folds. As we will detail below, we call such curves symmetry lines, and they represent the silhouettes $\mathcal{S}_\Omega$ in the domain (green curves in Figure 4c).

Since region (i), denoted here as $r_\Omega(i)$, folds on itself, it consists of two parts, one on either side of the symmetry line. Since each of these parts is by definition connected, and since both map continuously (due to $\mathbf{u}(\mathbf{x})$) to the connected region $r_\Psi(i)$ in the codomain, these two parts exhibit equivalent variation of value. That is, there is a continuous bijective spatial transformation in Ω that transforms one part to the other, such that their vector values match (notice that, for representing equivalence, the vector directions and magnitudes are not affected, i.e., not rotated, by this deformation).

Although we do not exploit these transformations explicitly in our current approach, they can be formulated for equivalent regions without symmetry as follows (in case of symmetry, one would need to treat the parts separated by symmetry lines as distinct regions in this consideration). Let $c^i_{\Psi\to\Omega}(r_\Psi)$ be the i-th correspondence from $c_{\Psi\to\Omega}(r_\Psi)$. Then,

$$\boldsymbol{\phi}^i_{r_\Omega}(\mathbf{x}) := \left\{ \boldsymbol{\xi} \mid \exists \boldsymbol{\xi} \in c^i_{\Psi\to\Omega}\left(c_{\Omega\to\Psi}(r_\Omega)\right) : \mathbf{u}(\boldsymbol{\xi}) = \mathbf{u}(\mathbf{x}) \right\} \quad (14)$$

transforms points $\mathbf{x} \in r_\Omega$ from r_Ω to the i-th equivalent region.

Besides folding of a single region, multiplicity can also be gen-erated by distinct regions in Ω that map on each other in Ψ. What we often observed are combinations thereof, i.e., regions that fold, and additional regions that map on the folded regions. The regions $(ii)_1$ and $(ii)_2$ in Figure 4a (green in Figure 4d and 4e) are an example for such a case. Both map to region (ii) in the codomain (Figure 4b), however, $(ii)_2$ folds along a symmetry line (green curve in Figure 4c) that goes from (6) to (10), i.e., each of the pairs (7)–(9) maps to a single point in Ψ. But at the same time, region $(ii)_1$ maps to region (ii) in Ψ too, without folding. Notice that there are eight regions in Ω that map to region (ii) in Ψ, with a resulting multiplicity of 10, i.e., two symmetry lines are involved here.

In general (including regions that do not exhibit a symmetry line), there is always a continuous bijective transformation that de-forms one equivalent region into another inside the domain, such that their multivariate values match. In other words, *equivalent re-gions* exhibit *equivalent variation* of multivariate value with respect to such a transformation. Put another way, each point in the do-main is contained in one equivalent region, and there are $\mu - 1$ ad-ditional points contained in the equivalent regions that exhibit the same value. Thus, equivalent regions cover the same multivariate values, and they exhibit the same multivariate variation under the abovementioned continuous transformation. Thus, they provide a concise representation of equivalence, which otherwise would need to be explored interactively in a tedious process (see also the video).

Concerning the role of $\mathcal{S}_\Omega$ as symmetry lines, remember that $\mathcal{S}_\Omega$ are zero-level isolines of the Jacobian determinant, and thus depict those loci where $\nabla \mathbf{u}(\mathbf{x})$ has at least one zero eigenvalue. This, on the other hand, means that there is an (eigenvector) direction in which the vector field $\mathbf{u}(\mathbf{x})$ does not vary. In traditional reflection
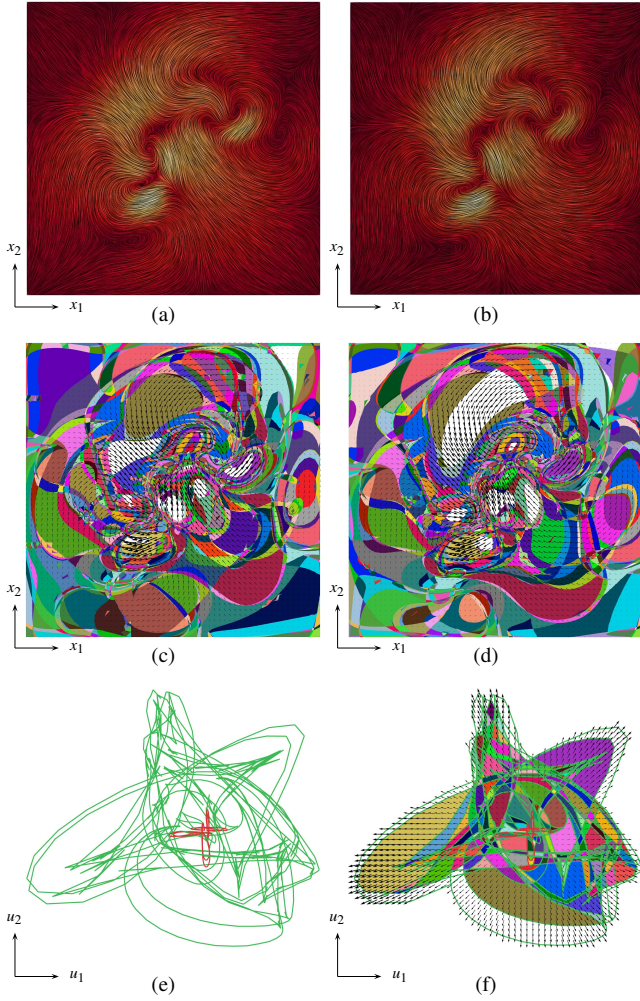
Figure 5: *Comparative visualization of mapping from Figure 2 ((a) and (c)), and a similar field ((b) and (d)). ((a) and (b)) LIC [CL93], with higher magnitudes by brighter colors (see (c) and (d) for flow orientation). Equivalent regions in domain ((c) and (d)), and in the common codomain (f), obtained from merged representation (e).*
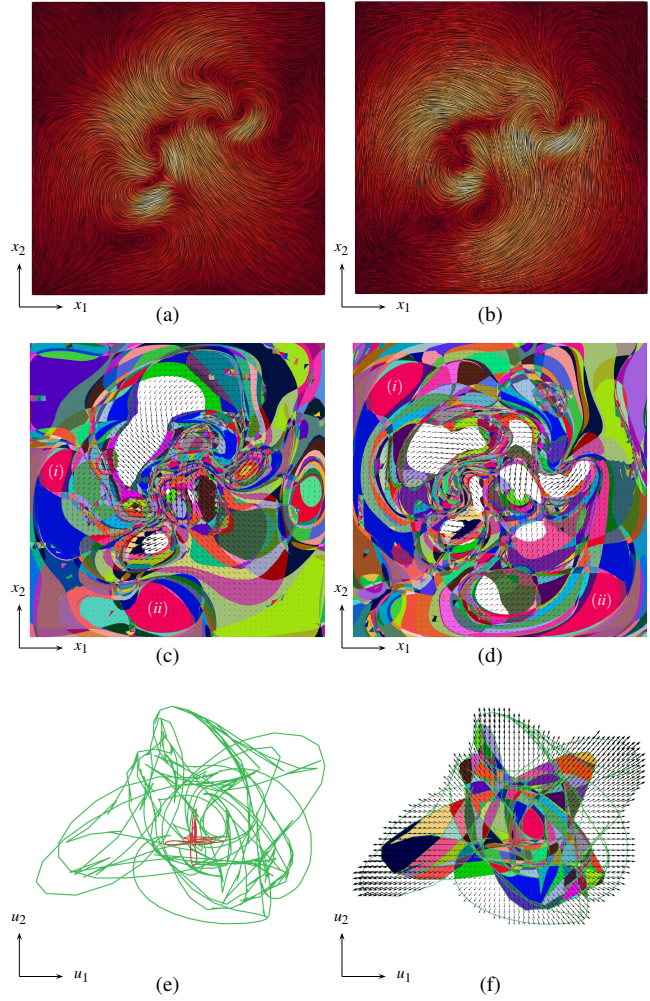
symmetry, the directional derivative of the mapping, in direction perpendicular to the symmetry line, is zero. In our case, the mapping is, in general, not angle-preserving, thus the zero directional derivative is generally not in direction perpendicular to $\mathcal{S}_\Omega$. But except for that, $\mathcal{S}_\Omega$ act as lines of reflection symmetry under $\mathbf{u}(\mathbf{x})$.

4.6. Comparative Visualization

Our approach lends itself also for comparative visualization of sets of mappings. To that end, one obtains, from each mapping $\mathbf{u}^i(\mathbf{x})$, the silhouette curves $\mathcal{S}^i_\Psi$ in the codomain, and merges them to the set $\mathcal{S}^\cup_\Psi$ (Figure 5e, green), as well as the boundary curves $\mathcal{B}^i_\Psi$ in the codomain, and merges those to $\mathcal{B}^\cup_\Psi$ (Figure 5e, red).

For $\mathbf{u}^i(\mathbf{x})$, the projected silhouettes $\overline{\mathcal{S}}^i_\Omega$ in the domain are:

$$\overline{\mathcal{S}}^i_\Omega := \left\{ \mathbf{x} \mid \mathbf{u}^i(\mathbf{x}) \in \mathcal{S}^\cup_\Psi \right\} \setminus \mathcal{S}^i_\Omega \tag{15}$$

Figure 6: *Same as Figure 5, but with more different field ((b) and (d)). Comparison of (f) with Figure 5f, and (c) and (d) with Figure 5c and 5d shows more white regions, i.e., less equivalence.*

(Equation 8), where $\mathcal{S}^i_\Omega$ are the silhouettes of $\mathbf{u}^i(\mathbf{x})$, in the domain. Accordingly, the projected boundaries $\overline{\mathcal{B}}^i_\Omega$ are in the domain:

$$\overline{\mathcal{B}}^i_\Omega := \left\{ \mathbf{x} \mid \mathbf{u}^i(\mathbf{x}) \in \mathcal{B}^\cup_\Psi \right\} \setminus \mathcal{B}^i_\Omega \tag{16}$$

(Equation 10), with the boundaries $\mathcal{B}^i_\Omega$ of $\mathbf{u}^i(\mathbf{x})$, in the domain.

In the codomain, the equivalent regions are obtained according to Section 4.4, but extracting the loops from $\mathcal{G}_\Psi := \mathcal{S}^\cup_\Psi \cup \mathcal{B}^\cup_\Psi$. In the domain, they are obtained by extracting the loops from $\mathcal{G}_\Omega := \overline{\mathcal{S}}^i_\Omega \cup \overline{\mathcal{B}}^i_\Omega \cup \mathcal{B}^i_\Omega$. The correspondences (Equation 12) that map from the domain of mapping i to the codomain are:

$$c_{\Omega^i \to \Psi}(r_{\Omega^i}) := \left\{ r_\Psi \mid \exists \mathbf{x} \in r_{\Omega^i} : \mathbf{u}^i(\mathbf{x}) \in r_\Psi \right\}, \tag{17}$$

(also here, $|c_{\Omega^i \to \Psi}(r_{\Omega^i})| = 1$). The correspondences (Equation 13) mapping from the codomain to the domain of mapping i are:

$$c_{\Psi \to \Omega^i}(r_\Psi) := \left\{ r_{\Omega^i} \mid \exists \mathbf{x} \in r_{\Omega^i} : \mathbf{u}^i(\mathbf{x}) \in r_\Psi \right\}, \tag{18}$$

and here too, $|c_{\Psi \to \Omega^i}(r_\Psi)| = \mu(\mathbf{u})$ with $\mathbf{u} \in r_\Psi$. The final step consists in removing those equivalent regions that are not present in all mappings (white regions in Figure 5c, 5d, and 5f).

In Figure 5, we compare two similar datasets, whereas Figure 6 shows the comparison of two vector fields that are more different. One can see that our approach directly transfers to the analysis of sets of mappings, i.e., the resulting equivalent regions represent value variation that is present in all vector fields. To effectively analyze these interrelations, we propose interactive techniques in Section 5.4, which are also demonstrated in the accompanying video.

Finally, we would like to draw the attention to the utility of the equivalent region representation in the codomain. As one can see, there is less white area in Figure 5f than in Figure 6f. That is, there is more equivalence in Figure 5f than in Figure 6f.

4.7. Discussion

Our approach is applicable to mappings of any dimension, with $\Omega \subset \mathbb{R}^n$, $\Psi \subset \mathbb{R}^m$, and $n = m$. In that sense, it complements that of Lehmann and Theisel [LT10], which addresses $n = m = 2$ and $n = 3, m = 2$. Similarly, it complements the works by Carr et al. [CGT*15] and Tierny and Carr [TC17], as outlined in Section 2.

For $n = m = 2$, Lehmann and Theisel obtain $\mathcal{S}_\Psi$ and $\mathcal{B}_\Psi$ in geometric representation, but do not extract the regions $\mathcal{R}_\Psi$ of uniform multiplicity therefrom. They obtain the multiplicity field by "raycasting" of the "folded manifold". To enable extraction of the uniform multiplicity regions both in the codomain and the domain (and therefrom the equivalent regions), however, a geometric representation of $\mathcal{R}_\Psi$ is mandatory, since the boundaries of uniform multiplicity regions in Ψ tend to be complex and involve thin regions. A pixel-based connected component labeling of the raycast multiplicity field could therefore not provide the correct result. However, as detailed below, obtaining a geometric representation of $\mathcal{R}_\Psi$ is computationally costly.

To make an exact extraction feasible, we restrict our implementation to mappings given on simplicial complexes that are path-connected, because a simplex remains a simplex under arbitrary deformation of its vertices, and thus barycentric interpolation of vector value in Ω carries over to barycentric interpolation of position in Ψ (Section 5). We also obtain $\mathcal{R}_\Omega$ in geometric representation, although the cost of its computation increases even faster with dataset size and data complexity. We therefore provide a shader-based alternative for visualization of $\mathcal{R}_\Omega$, which we use when computation times become too long. The advantage of the fully geometric approach is, however, that it is exact and that one can provide the results in geometry format, which is highly useful for investigating small and complex equivalent regions.

5. Algorithm

As just mentioned, we assume the data grid of the mapping(s) $\mathbf{u}(\mathbf{x})$ to be a simplicial complex (which, in case of comparative visualization, can differ among the fields), interpolated with barycentric interpolation, and that no two nodes of the grid(s) exhibit identical n-variate value. In practice, this is usually ensured by perturbation or simulation of simplicity [EM90]. As a consequence, a simplex
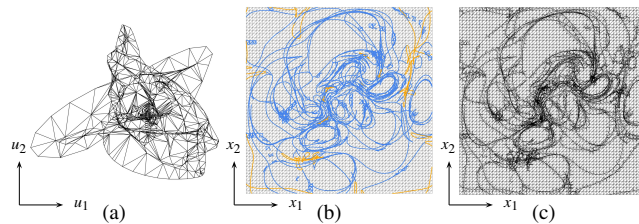

Figure 7: *Mapping from Figure 2. (a) Smallest closed loops extracted from silhouettes and boundaries in codomain split at intersections, and each loop triangulated, provides regions in codomain (Figure 3f). (b) Projected silhouettes, projected boundaries, and edges of original grid, in domain. (c) Triangulation of (b) and computing correspondences provides regions in domain (Figure 3e).*

Δ_Ω in Ω maps to a simplex Δ_Ψ in Ψ, remains by nature convex, and can only change handedness. Another benefit of requiring simplicial grids is that $\mathcal{S}_\Omega$ and $\mathcal{B}_\Omega$ consist of faces (i.e., edges in case of triangular grids, and triangles in case of tetrahedral grids) of the original grid (Figure 2e), and thus $\mathcal{S}_\Omega$ and $\mathcal{S}_\Psi$, as well as $\mathcal{B}_\Omega$ and $\mathcal{B}_\Psi$, are piecewise linear. Higher-order interpolation functions, in contrast, would induce complex isocontour geometry, which would be hard to represent and work with.

5.1. Silhouettes and Boundaries

The first stage of our algorithm traverses all faces of the simplicial complex and determines if they are part of a silhouette or boundary. If the face is adjacent to only one simplex, it is part of a boundary. If, on the other hand, the two simplices adjacent to the face have different sign of the Jacobian determinant $\det \nabla \mathbf{u}(\mathbf{x})$ (which is constant per simplex), the simplices are opposite-handed and thus the face represents part of a silhouette. The silhouette faces and boundary faces are mapped to the codomain by looking up the vector values at their vertices, providing $\mathcal{S}_\Psi$ and $\mathcal{B}_\Psi$. Subsequently, they are intersected with each other, i.e., they are split at intersections. As detailed in Section 4.3, the resulting set is closed. Figure 2d shows an example for our $n = 2$ implementation.

5.2. Equivalent Regions in Codomain

Next, the regions of uniform multiplicity $\mu(\mathbf{u})$ are obtained by extracting the smallest closed manifolds from $\mathcal{S}_\Psi \cup \mathcal{B}_\Psi$ (Section 4.4), and each of these regions is triangulated and given a unique ID. Figure 7a shows an example of the resulting mesh for our $n = 2$ implementation, and Figure 3f the color-coded equivalent regions.

5.3. Equivalent Regions in Domain

To obtain the regions with uniform multiplicity in the domain, we need the projected silhouettes and projected boundaries. To this end, we first (conceptually) transform the data grid to the codomain (Figure 2d). That is, we transform the simplices Δ_Ω from the domain to simplices Δ_Ψ in the codomain. We employ dual representation, i.e., the "positions" of the vertices of Δ_Ψ are the original vector values $\mathbf{u}$, and the "values" at the vertices of Δ_Ψ are the original positions $\mathbf{x}$. We then traverse all silhouette faces and boundary faces,
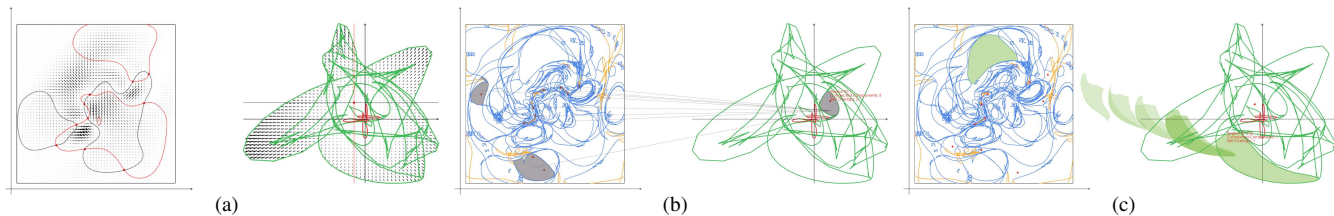
Figure 8: *Interaction techniques. Domain (left) and codomain (right). (a) Point-based exploration, including isolines. (b) Region-based exploration. (c) Mapping between domain and codomain by animation, revealing folding etc. (see accompanying video).*

as determined in Section 5.1, and split each of them with all Δ_Ψ, i.e., if such a face intersects a Δ_Ψ, we determine the part of the face that is located inside the simplex. We then obtain the vertex positions of this face part by interpolating the "position values" within Δ_Ψ, which gives their position in the domain, and thus results in the projected silhouettes and projected boundaries in the domain.

To support consistent interpolation, we throw these projected silhouette parts and boundary parts together with the vertices and edges of the original data grid (Figure 7b), and perform constrained triangulation [Kor] (Figure 7c). We then populate the resulting triangulation with the mapping $\mathbf{u}(\mathbf{x})$, i.e., we copy the node values and interpolate $\mathbf{u}$ for the new vertices. Since barycentric interpolation is linear, the resulting field is, up to floating-point accuracy, identical to the original one. Subsequently, we traverse each of the simplices of the triangulation, interpolate $\mathbf{u}$ at their center, and look up the ID from the mesh in the codomain (Section 5.2 and Figure 7a), resulting in the equivalent regions, as shown in Figure 3e.

5.4. Interaction Techniques

There are several difficulties with static representation of equivalence in mappings. As can be seen, e.g., from Figure 4, color-coding of a rather large number of equivalences is difficult due to color perception limitations. Second, even if the colors could be clearly distinguished and the regions large enough, one needs to search the correspondences visually, e.g., between Figure 4a and 4b. Drawing (bundled) edges to represent the correspondences would, due to their large number, also lead to perception issues. Furthermore, for a complete picture, one needs, at the same time, have the multiplicity information present (this is why we are showing color-codings of multiplicity together with the equivalent region representations in this paper). And last but not least, one also needs to observe the mapping, which we achieve in the static representations by superposition with arrow glyphs.

Certainly, the amount of information is too large for an effective static representation. This is why we present here some approaches for interactive exploration. We refer the reader to the video that accompanies this paper for a demonstration.

Point-based exploration. The most basic techniques to explore equivalence in mappings are (i) to move the mouse in the domain and interactively show the corresponding moving point in the codomain, or (ii) move the mouse in the codomain and show the equivalent moving point(s) in the domain. It is particularly useful to provide $\mathcal{S}_\Psi$ and $\mathcal{B}_\Psi$ in the codomain for context in the latter

mode, since crossing $\mathcal{S}_\Psi$ causes "bifurcations" of the points in the domain, i.e., they annihilate or originate in pairs, while crossing $\mathcal{B}_\Psi$ makes single points appear or disappear. Optionally, if the mouse is at position $\mathbf{u} =: (u_1, \ldots, u_n)^\top$ in the codomain, displaying the n isocontours $\{\mathbf{x} \mid \mathbf{u}(\mathbf{x})_i = u_i\}$ in the domain provides context about the influence of $\mathcal{S}_\Psi$ and $\mathcal{B}_\Psi$, and the relation between the components of the mapping. Notice that such isolines represent axis-aligned straight lines in the codomain. Figure 8a exemplifies the point-based exploration.

Region-based exploration. Interactive point-based exploration of course cannot provide the full picture of equivalence. Thus, the straightforward extension is to do interactive region-based exploration. Again, the user can move the mouse (i) in the domain and observe which equivalent region is interactively highlighted in the codomain, or (ii) move the mouse in the codomain and observe which equivalent region(s) are highlighted in the domain. Optionally, straight lines connecting the center of the region in the codomain with the center(s) of the equivalent region(s) in the domain can be shown, to support identification of small regions. Figure 8b exemplifies this approach. Notice that this mode is very useful during comparative visualization, in which case the lines point to the domains of all mappings under consideration.

Transformation of regions. Given that correspondence in terms of regions has been reasoned out, the user might want to understand the detailed mapping of regions, in terms of deformation and folding. For such questions, the user can click a region in the domain or the codomain, which starts an animation that transforms the region(s) respectively. Clicking in the codomain makes the region(s) move from the codomain to the domain (or domains in case of comparative visualization), and clicking in the domain makes the respective region(s) transform form the domain to the codomain. Optionally, both views are centered at the region(s) under consideration. See Figure 8c for an example. The animation is achieved by linear interpolation (where the interpolation parameter is "animation time") between position $\mathbf{x}$ and value $\mathbf{u}$ of each vertex of the underlying mesh.

6. Results

We demonstrate the utility and interpretation of our approach using two cases from computational fluid dynamics (Sections 6.1 and 6.2), and a bivariate field from climatology (Section 6.3). Performance details are given in Table 1, and discussed in Section 6.4.

6.1. Convective Flow

The 2D vector field that we used in the previous sections represents an axis-aligned cross section from a 3D computational fluid dynamics (CFD) simulation of airflow in a closed container, driven by heat-induced convection, with no-slip boundary conditions. A common research question in such configurations is the variation of the airflow subject to the distance from a wall, which is often examined based on cross-sectioning with varying distance from the boundary, followed by projection of the resulting vectors onto the section, resulting in 2D vector fields.

Figures 2a and 5a provide an overview of the vector field, Figure 2b of its codomain, Figure 4a shows the equivalent regions in the domain and Figure 4b in the codomain, and the respective multiplicities are provided in Figures 3c and 3d. If we look at the deformed domain manifold in the codomain (Figure 2d), we can see that the CFD solver did not perfectly meet the no-slip boundary conditions, i.e., one can see that the red boundary curve spreads around the zero velocity point in Ψ (see, e.g., Figure 8a (right) for the location of the origin of the codomain). The deformed domain manifolds (Figure 2d) also find a use for judging the discretization of vector fields: large cells in the codomain represent high relative variation, i.e., large gradients compared to the grid resolution in the domain, which might represent aliasing. This is consistent with strong velocity variation in the red region (*iii*) in Figure 4a and 4b.

Having a look at the multiplicity field in the domain (Figure 3c), we identify four prominent regions with low multiplicity, marked (*i*), (*iii*), (*iv*), and (*v*) in Figure 4a and 4b. From Figure 4b, we see that all these regions are located at the boundary of the codomain of the vector field, and from Figure 2f we see that those regions are all confined by a silhouette curve there, which means they all have to exhibit multiplicity two due to symmetry. The fact that they are located at the boundary of the codomain also tells us that they are among the fastest regions, since radius in the codomain represents velocity magnitude. In this case, the prominent unique (low multiplicity) regions identified by our approach represent coherent flow regions with high velocity.

To investigate the variation of the projected 2D field with respect to distance from the boundary in the underlying 3D flow, we employ comparative visualization of this slice with another slice that is close (Figure 5), and with a slice that is farther away (Figure 6). In Figure 5, one can see that many equivalent regions in the left part of the domain are spatially consistent in both fields, indicating strong similarity, i.e., in terms of value *and* position. This conforms to the visual inspection of Figure 5a and 5b, which shows that velocity direction differs in particular in the right lower part of the domain. The large white region in Figure 5d tells us, on the other hand, that although vector field direction is very similar in both fields in this region, the magnitude differs, leading to velocities in the second field that are not present in the first one. In the codomain, there are more white regions in Figure 6f than in Figure 5f, indicating less equivalence in the more distance slice pair, and inspection of Figure 6c and 6d shows that there are only few equivalent regions that are located at similar positions, indicating less positional equivalence. Nevertheless, in both Figure 6c and 6d, we observe, e.g., two large regions ((*i*) and (*ii*)), depicting equivalence between these two fields with respect to rather slow top-right flow (see arrow glyphs).
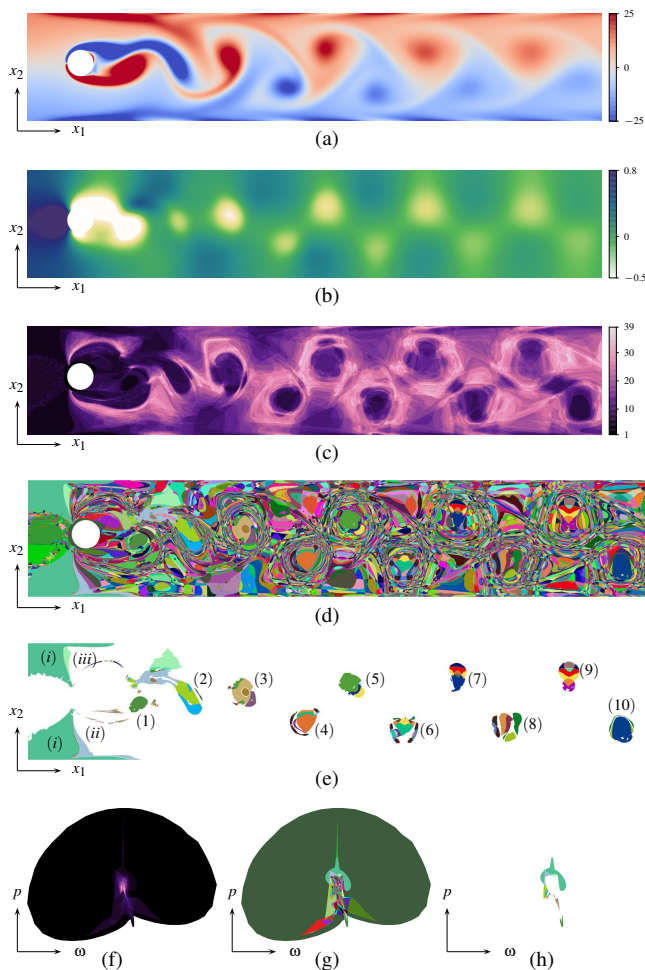

Figure 9: *Kármán flow around obstacle (white disc). Vorticity ω(x) (a) and pressure p(x) (b) in domain. Multiplicity in domain (c) and codomain (f). Equivalent regions in domain (d) and codomain (g). Selected regions in domain (e) and codomain (h).*

6.2. Kármán Flow

Let us now investigate the interplay between vorticity $\omega(\mathbf{x}) := \nabla \times \mathbf{u}(\mathbf{x})$ and pressure $p(\mathbf{x})$ in a time step of a 2D CFD simulation exhibiting vortex shedding (notice that in 2D flow fields, vorticity is a scalar field). As we can see from Figure 9a and 9b, there is some correlation between vorticity and pressure within vortices. In this case, low multiplicity regions (Figure 9c) reveal the vortices in terms of uniqueness. Furthermore, the equivalent regions (Figure 9d) are comparably large within the vortices. To assess the similarity of the vortices, we investigate their equivalent regions (Figure 9e). Since this would be usually accomplished by interactive exploration, we provide here the regions selected during our interactive exploration.

Interestingly, the leftmost green equivalent regions (*i*) are symmetric, whereas the remaining regions are rather antisymmetric due to the oscillating vortex shedding dynamics. The equivalent regions reveal that vortices (7) and (9) are most similar in terms of bivariate
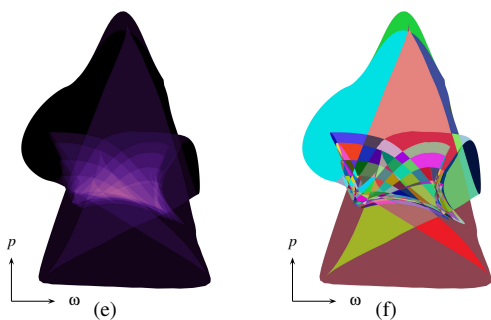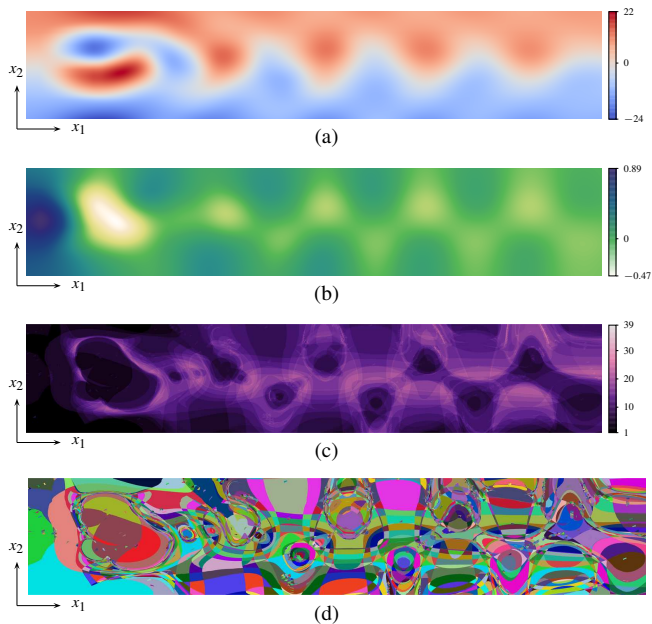
Figure 10: *Case from Figure 9, but smoothed with σ = 6. Vorticity (a), pressure (b), multiplicity (c), and equivalent regions (d) in domain. Multiplicity (e) and equivalent regions (f) in codomain.*

Figure 11: *Investigation of robustness by application to time series of case from Figure 10. (c) Same as Figure 10d, together with preceding time step (b), and time step (a) preceding that step.*

variation, i.e., their dark blue, red, and yellow regions are equivalent. Similarly, vortices (7) and (5) share the yellow and medium blue regions, and vortices (5) and (3) the green one. We identify decreasing similarity (smaller and less equivalent regions in the domain) of adjacent vortices, as we approach the obstacle. In the lower row of the vortices, we identify vortex (10) and (8) sharing the dark and light green regions, vortices (8) and (6) sharing the brown, dark green, and red regions, and vortices (6) and (4) sharing the green, purple, and dark brown regions. Vortex (2) is different, it shares the green and blue regions with the area behind the obstacle, and this area, in turn, shares the light blue and light green areas (*ii*) and (*iii*) in front of the obstacle. Also vortex (1) is different—it has a unique region, i.e., all of its bivariate values are unique in the dataset. Finally, the large blue region in vortex (10) does not have an equivalent in the other vortices, as well.

Identifying all these equivalences manually in the codomain, without the help of equivalent regions, would be very tedious, since the regions are thin and hard to determine. Our example also demonstrates the utility of equivalent regions as "building blocks
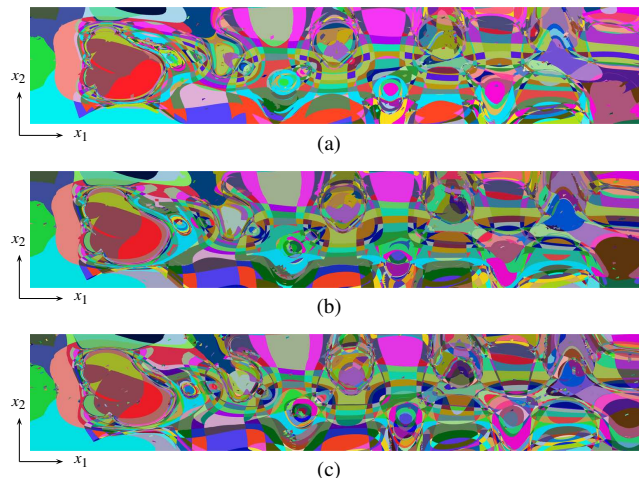
for multivariate brushing", in addition to their power of identifying equivalence. That is, traditional interactive exploration without the help of equivalent regions would require much more efforts to provide an overview of the characteristics of this example.

As discussed in Section 3 and apparent in Figure 9d, datasets with high bivariate variation tend to produce many small equivalent regions. Although these regions are essential for small-scale analysis of equivalence, they do not provide a clear overview in such cases. To obtain a representation with respect to large-scale equivalence, we motivated in Section 3 smoothing of the bivariate field prior to equivalence visualization. Thus, we employed Gaussian smoothing with σ = 6 to the Kármán dataset (Figure 10), i.e., independently to both the vorticity and pressure field. Since the original data was given on a triangular mesh, we resampled it on a uniform grid for smoothing, assigning zero in the obstacle region.

We observe two main effects of this smoothing. On the one hand, the multiplicity field (Figure 10c) has overall lower values (although the maximum stays 39), on the other hand, in the domain, the equivalent regions become larger, in particular in the regions between the vortices (Figure 10d). In the codomain, both the multiplicity field (Figure 10e) and the equivalent regions (Figure 10f) take on a more regular structure, also in time. Interestingly, in the smoothed data, the vortices are identified as more compact regions, but because the properties have been "merged", there is less equivalence between adjacent vortices. On the other hand, the areas between the vortices exhibit stable large-scale equivalent structure, alternating vertically, consistent with vortex shedding dynamics.

Finally, to investigate the robustness of our approach, we applied it to subsequent time steps of the Kármán simulation smoothed with σ = 6 (Figure 11). After the equivalent regions have been extracted, we color-coded the region IDs of the first time step in this series, and determined the color-coding of the IDs of the subsequent time steps by spatial similarity, to make the results better comparable. One can see that our approach is quite robust in space and time.
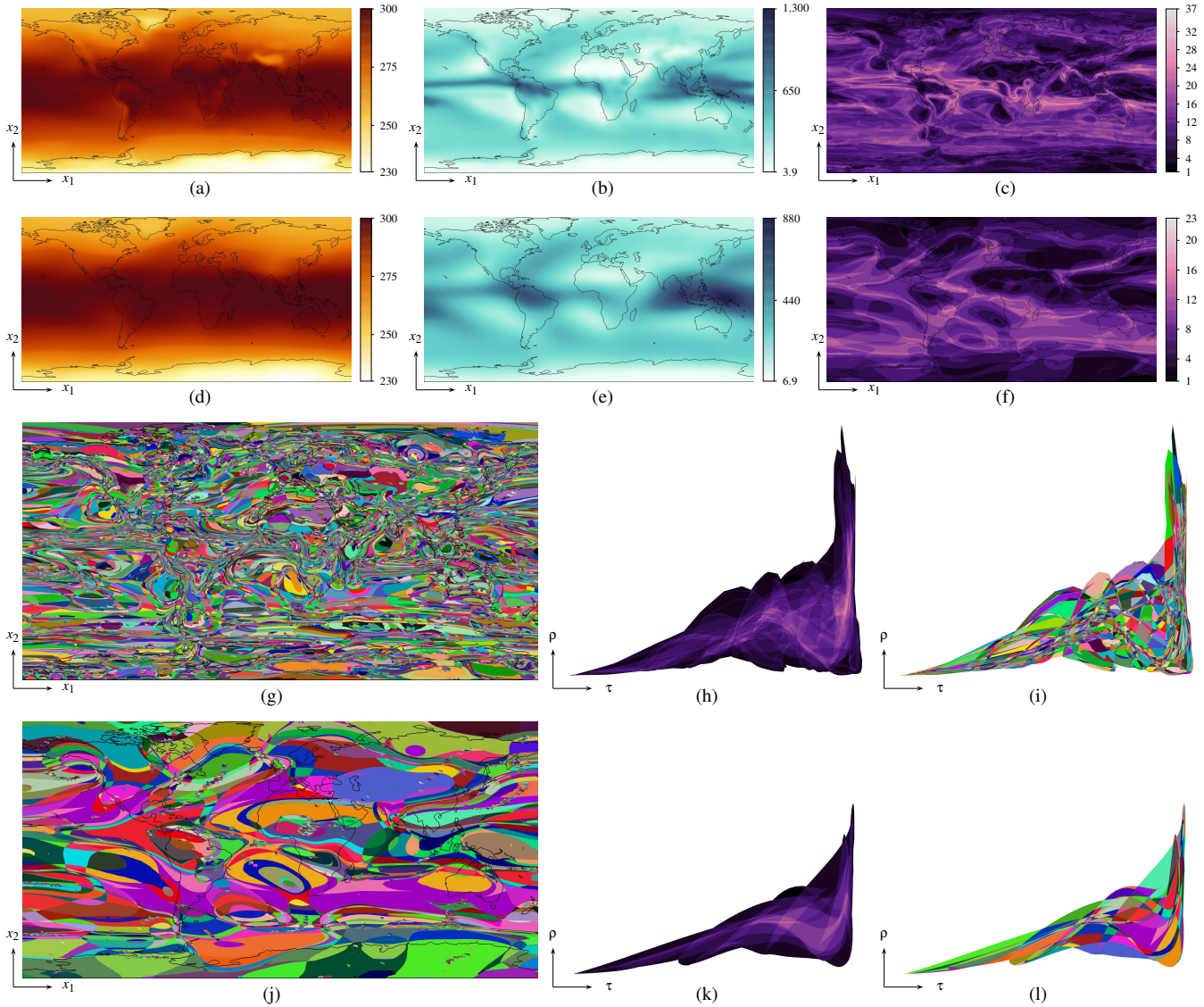
Figure 12: *Climate Zones example based on averaged temperature and precipitation from 1979 until 2015, with Gaussian smoothing with $\sigma = 4$ ((a)–(c) and (g)–(i)), and with $\sigma = 14$ ((d)–(f) and (j)–(l)). Temperature $\tau(\mathbf{x})$ ((a), (d)), precipitation $\rho(\mathbf{x})$ ((b), (e)), multiplicity in domain ((c), (f)), equivalent regions in domain ((g), (j)), multiplicity in codomain ((h), (k)), and equivalent regions in codomain ((i), (l)).*

6.3. Climate

We now demonstrate the utility of our concept for climate analysis. A common approach in climate research is to classify different climate types, with the classification system due to Köppen and Geiger [Gei54] being often employed. This classification system is quite complex, taking into account precipitation and temperature patterns, such as monthly-averaged temperature and precipitation, yearly-averaged temperature, seasonal precipitation, coldest month, etc. The result of this classification is shown in Figure 13d.

We computed a 37-year average temperature and average precipitation based on MERRA's [RSG*11] monthly-averaged data from January 1979 to December 2015. To avoid "over-segmentation", we applied Gaussian smoothing to the resulting fields with $\sigma = 4$

(Figures 12a and 12b), and $\sigma = 14$ (Figures 12d and 12e). Whereas $\sigma = 4$ provides very many small regions (Figure 12g), which reveal fine-scale equivalence of climate across the globe, the version with $\sigma = 14$ gives a more large-scale picture (Figure 12j and 13b). Typically, one would start an investigation using our approach with such a large-scale representation to understand global trends, followed by detailed analysis with the fine-scale representation. Here, we exemplify the investigation at the large scale, i.e., $\sigma = 14$.

To demonstrate the effectiveness of our approach, we measured the total area of all regions in the domain that correspond to a certain region ID, i.e., that represent equivalent regions. We then sorted the region IDs with respect to that total area and selected the 25 largest, providing the subset shown in Figure 13a, shown with 25
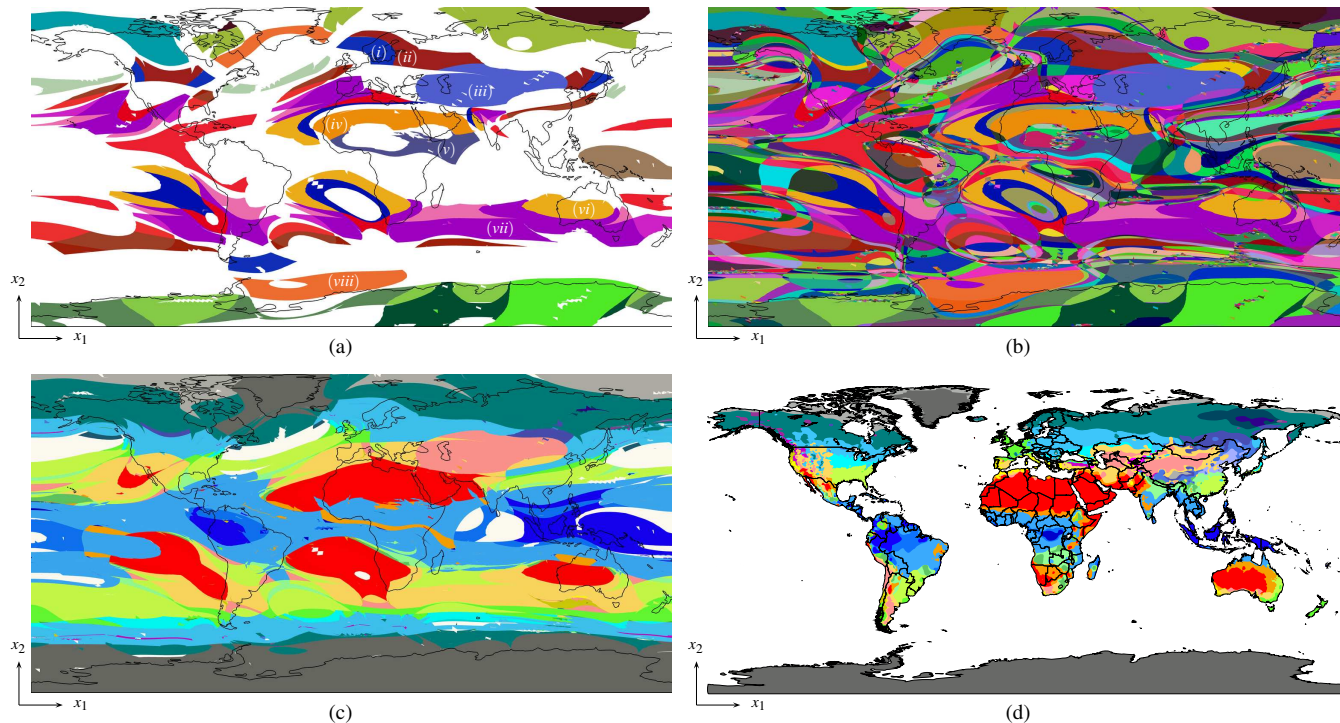
Figure 13: *Case from Figure 12. To show the effectiveness of our approach, we selected the 25 equivalent region IDs with largest area in the domain (a). (b) same as Figure 12j, for comparison. (c) Assignment of the color codes from Köppen and Geiger's classification (d) [VPFM] to our regions from (b) reveals strong correlation between equivalent regions and the regions defined by that complex classification system.*

different colors. This leads to several observations. Region (*iii*) captures the deserts in Asia, including Karakum, Taklamakan, Gobi, and Ordos. This region has multiplicity one and corresponds well to the respective skin-colored area in Köppen and Geiger's classification in Figure 13d. Central Sahara (region (*iv*)) is also captured by a single unique region, extending to the Arabian desert, and corresponding to a large part of the red region in Köppen and Geiger's classification. Another large and unique region (*v*) captures the region south of the Sahara, corresponding well to Köppen and Geiger's classification. The arid region (*vi*) in the center of Australia also fits the classification well, however, this region is not unique in our result, it corresponds to the arid region in south Africa. Region (*vii*), on the other hand, corresponds to humid subtropical climate according to Köppen and Geiger's classification, and links south east Australia with south east Africa, central east Brazil, the south of the United States, south west Europe, and north India, all conforming to Köppen and Geiger's classification. Regions (*i*) and (*ii*), on the other hand, correspond to the climate in Scandinavia, the border between the United States and Canada, and the region north west from Japan. Finally, region (*viii*) is the only region among the selected ones that links the antarctic region south of South America with south Greenland. Despite the overall tendency toward north–south symmetry, we were surprised that the northern and southern polar regions do not exhibit mutual equivalence. This might relate to the different ocean–land proportions.

As a final step, we assigned (in analogy to Figure 11) the spatially best-matching colors from Köppen and Geiger's classification

(Figure 13d) to all of our equivalent regions with $\sigma = 14$, resulting in Figure 13c, indicating the consistency between our result and this classification system. It has to be noted, however, that one cannot expect that our results are identical, because Köppen and Geiger's classification rules are complex and involve various temperature and precipitation data, whereas we have only used a single long-term averaged temperature and precipitation field. Nevertheless, we think that our experimental results demonstrate a promising direction of research, advocating the application of our approach in climate research—in particular, since our result is parameter-free, whereas Köppen and Geiger's classification system consists of a complex set of rules with many parameters and data fields.

Table 1: *Dataset sizes, and timings (computations in codomain Ψ / domain Ω), measured on an Intel Core i5-6300HQ CPU (2.30 GHz). Timings are only given for the CPU-based implementation. The shader alternative, which replaces the computations in Ω, takes 0.51 s for refresh for the Kármán dataset.*

Dataset	Cells	Vertices	Proc. Time (Ψ / Ω)
Convective Flow	6 272	3 249	1.79 s / 36.75 s
Kármán	34 422	17 552	144 s / 25 310 s
Kármán ($\sigma = 6$)	20 000	10 251	23 s / 1 864 s
Climate ($\sigma = 4$)	24 120	12 285	203.6 s / 24 574 s
Climate ($\sigma = 14$)	24 120	12 285	24.7 s / 1 188 s

6.4. Performance

Table 1 provides the runtimes of our CPU-based implementation. Calculations in the domain Ω are dominated by the intersection operations of the projected silhouettes and projected boundaries (Section 5.3), which we solve naively at the moment. The acceleration of this procedure is, however, nontrivial, as the edges to intersect typically span large parts of Ω. Our alternative shader-based approach (Section 4.7) results in significant speedups (over 500 000 times on the Kármán dataset), taking 0.51 s for a refresh at a resolution of 1920×1080 pixels. Notice that such a refresh takes place only when the user zooms or pans, all other interactions take place at very high frame rates (see the accompanying video).

7. Conclusion

We presented the novel concept of equivalence in nD n-variate mappings, and derived therefrom the concept of equivalent regions in such mappings. We identified the critical lines by Lehmann and Theisel [LT10] as those lines separating regions of uniform multiplicity in the codomain, and extracted those regions geometrically. Correspondingly, we introduced the concept of projected silhouettes and projected boundaries in the domain, and have shown that those separate regions of uniform multiplicity in the domain. We extracted the respective regions also geometrically in the domain, and established correspondence between the regions in the domain and those in the codomain, providing our equivalent regions. We then extended our approach for comparative visualization, and presented a set of interactive visualization techniques for the exploration of equivalence. To reduce computation times in case of large and complex data, we proposed an alternative shader-based visualization of the regions in the domain. Despite the substantial acceleration provided by this, our approach is still computationally costly, and future work should investigate further approaches for acceleration. Also, although our concept is applicable to arbitrary dimension n, advanced acceleration strategies become essential already for $n = 3$, which we would like to address as future work.

Acknowledgments

References

[BW08] BACHTHALER S., WEISKOPF D.: Continuous scatterplots. *IEEE Transactions on Visualization and Computer Graphics 14*, 6 (2008), 1428–1435. 1

[CD13] CARR H., DUKE D.: Joint contour nets: Computation and properties. In *Proceedings of the IEEE Pacific Visualization Symposium* (2013), pp. 161–168. 1

[CD14] CARR H., DUKE D.: Joint contour nets. *IEEE Transactions on Visualization and Computer Graphics 20*, 8 (2014), 1100–1113. 1

[CGT*15] CARR H., GENG Z., TIERNY J., CHATTOPADHYAY A., KNOLL A.: Fiber surfaces: Generalizing isosurfaces to bivariate data. *Computer Graphics Forum 34*, 3 (2015), 241–250. 1, 7

[CL93] CABRAL B., LEEDOM L. C.: Imaging vector fields using line integral convolution. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques* (1993), pp. 263–270. 6

[DCK*12] DUKE D., CARR H., KNOLL A., SCHUNCK N., NAM H. A., STASZCZAK A.: Visualizing nuclear scission through a multifield extension of topological analysis. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2033–2040. 1

[EHP08] EDELSBRUNNER H., HARER J., PATEL A. K.: Reeb spaces of piecewise linear mappings. In *Proceedings of the Twenty-fourth Annual Symposium on Computational Geometry* (2008), pp. 242–250. 1

[EM90] EDELSBRUNNER H., MÜCKE E. P.: Simulation of simplicity: A technique to cope with degenerate cases in geometric algorithms. *ACM Transactions on Graphics 9*, 1 (1990), 66–104. 7

[Gei54] GEIGER R.: *Klassifikation der Klimate nach W. Köppen*. Springer, 1954, pp. 603–607. 11

[HH89] HELMAN J., HESSELINK L.: Representation and display of vector field topology in fluid flow data sets. *IEEE Computer 22*, 8 (1989), 27–36. 2

[HW09] HEINRICH J., WEISKOPF D.: Continuous parallel coordinates. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 1531–1538. 1

[Kor] KORNBERGER B.: Geom Software: The Fade 2D and 2.5D C++ library. http://www.geom.at/products/fade2d. 8

[LC87] LORENSEN W. E., CLINE H. E.: Marching cubes: A high resolution 3D surface construction algorithm. In *Proceedings of ACM SIGGRAPH Computer Graphics* (1987), vol. 21, pp. 163–169. 2

[LT10] LEHMANN D. J., THEISEL H.: Discontinuities in continuous scatter plots. *IEEE Transactions on Visualization and Computer Graphics 16*, 6 (2010), 1291–1300. 1, 3, 4, 7, 13

[PSBM07] PASCUCCI V., SCORZELLI G., BREMER P.-T., MASCARENHAS A.: Robust on-line computation of Reeb graphs: Simplicity and speed. *ACM Transactions on Graphics 26*, 3 (2007). 1

[RSG*11] RIENECKER M. M., SUAREZ M. J., GELARO R., TODLING R., BACMEISTER J., LIU E., BOSILOVICH M. G., SCHUBERT S. D., TAKACS L., KIM G.-K., BLOOM S., CHEN J., COLLINS D., CONATY A., DA SILVA A., GU W., JOINER J., KOSTER R. D., LUCCHESI R., MOLOD A., OWENS T., PAWSON S., PEGION P., REDDER C. R., REICHLE R., ROBERTSON F. R., RUDDICK A. G., SIENKIEWICZ M., WOOLLEN J.: MERRA: NASA's modern-era retrospective analysis for research and applications. *Journal of Climate 24*, 14 (2011), 3624–3648. 11

[TC17] TIERNY J., CARR H.: Jacobi fiber surfaces for bivariate Reeb space computation. *IEEE Transactions on Visualization and Computer Graphics 23*, 1 (2017), 960–969. 2, 7

[VPFM] VON PEEL M. C., FINLAYSON B. L., MCMAHON T. A.: Enhanced, modified, and vectorized by Ali Zifan. - Hydrology and Earth system sciences: "Updated world map of the Köppen-Geiger climate classification" (supplement). University of Melbourne. CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=47086877. 12