DOI: 10.1111/j.1467-8659.2012.03112.x
Eurographics Conference on Visualization (EuroVis) 2012
S. Bruckner, S. Miksch, and H. Pfister
(Guest Editors)

Volume 31 (2012), Number 3

Comparative Visual Analysis of 2D Function Ensembles

H. Piringer,[1] S. Pajer,[1] W. Berger,[1] and H. Teichmann[2]

[1]VRVis Research Center, Vienna, Austria
[2]AVL List GmbH, Graz, Austria

Abstract

In the development process of powertrain systems, 2D function ensembles frequently occur in the context of multi-run simulations. An analysis has many facets, including distributions of extracted features, comparisons between ensemble members and target functions, and details-on-demand. The primary contribution of this paper is a design study of an interactive approach for a comparative visual analysis of 2D function ensembles. The design focuses on a tight integration of domain-oriented and member-oriented visualization techniques, and it seeks to preserve the mental model of 2D functions on multiple levels of detail. In this context, we propose a novel focus+context approach for visualizations relying on data-driven placement which is based on labeling. We also extend work on feature-preserving downsampling of 2D functions. Our design supports a comparison of 2D functions based on juxtaposition, overlay, and explicit differences. It also enables an analysis in terms of extracted scalar features and 1D aggregations. An evaluation illustrates a workflow in our application context. User feedback indicates a time saving of 70% for common tasks and a qualitative gain for the entire development process.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

Multi-run simulations are important for studying complex systems in science and engineering [Hel08]. Depending on the type of simulation, results may have different characteristics. Analyzing ensembles of scalar results is possible using standard multivariate visualization techniques [BPFG11]. Much work addresses the visual analysis of many 1D functions such as time-series [HS04, KMG*06, M*08]. However, ensembles of higher-dimensional results are increasingly challenging to analyze.

This paper addresses the visual analysis of ensembles of 2D functions, which are essential in a variety of applications, e.g., slicing volumetric data [NFB07]. The application background motivating this work is the development process of powertrain systems by means of multi-run simulations as described by Matkovic et al. [MGKH09]. In this context, 2D functions are a common type of simulation results. An example is to simulate the maximal pressure that may occur within a circular bearing – angle and bearing width defining the 2D function domain. Such data is essential to analyze at which load and how bearings might break [Off11]. The figures of this paper refer to this example which is described in more detail in Sec. 8.1. Another application of 2D functions is to analyze the sensitivity of a scalar simulation result (e.g., generated torque) to conditions varying during the operation of an engine. In this case, operational parameters (typically "engine speed" and "load") define the domain of the 2D function, and they are varied independently from other simulation parameters.

1.1. Tasks

Engineers for optimizing powertrain systems face different tasks in the context of 2D function ensembles. Based on predesign studies by interviewing domain experts and by observations with contextual inquiries [SIBB11], we identified tasks on different levels of abstraction [Mun09]. On a high level of abstraction, engineers need to:

- **Assess data quality** – are simulation results plausible?
- **Analyze sensitivities** – which simulation parameters affect the 2D function ensemble and how?
- **Optimize engine designs** – which ensemble members behave desirable?

On a low level of abstraction, tasks include to:

- **Assess distributions of features.** Depending on the aggregation, this can be member-oriented or domain-oriented. For example, a member-oriented distribution

considers the maximum for each 2D function, while a domain-oriented distribution considers the maximum of the entire ensemble for each point of the 2D domain.
- **Compare.** Comparison is necessary for characterization (e.g., compare one ensemble member to another one) and for optimization (e.g., compare to a target 2D function).
- **Retrieve details-on-demand.** Details may refer to groups of members (e.g., clusters), single members, and specific sample values.

The domain experts described these tasks as cumbersome and time-consuming. Current tools fail to support transitions between overview and detail. Overviews are restricted to common diagrams (e.g., scatterplots) showing pre-computed scalar statistics. Details involve 2D or 3D plots of single ensemble members and are inspected one-by-one, where "*discriminating differences depends on the trained eye of experts*", as one engineer said. As a consequence, visualizations of 2D functions are currently rather used for presentation than for analysis which was described as a significant source of uncertainty in the process.

1.2. Design goals

Based on these observations, we defined the subsequent list of design goals for an interactive visual approach for the analysis of 2D function ensembles:

- **G1**: Support a flexible exploration of different types of member-based and domain-based overviews.
- **G2**: Enable a seamless access to details for a subset of ensemble members or a single 2D function.
- **G3**: Facilitate comparisons between ensemble members.
- **G4**: Provide quantitative results where possible.
- **G5**: Support an analysis of local features within the 2D function domain.
- **G6**: Preserve the mental model "2D function ensemble" on all levels-of-details.
- **G7**: Scale to 2D functions with many samples.
- **G8**: Scale to hundreds of 2D functions at interactive rates.
- **G9**: Achieve a seamless integration in their workflow.

Throughout three years, we iteratively refined design concepts based on paper drawings and increasingly functional software prototypes. Four domain experts ranging from engineer to management level participated in this process.

1.3. Contributions

The primary contributions of this paper are (1) a *design study* of an interactive approach for a comparative visual analysis of 2D function ensembles, and (2) an *evaluation* of the approach based on an exemplary workflow in simulation-based engine design and the report of user feedback. Secondary contributions include (3) an adaptation of an algorithm for *extrema-preserving downsampling* to our needs, and (4) a novel *focus+context approach for glyph-based visualization* techniques which is based on labeling.

2. Related Work

The analysis of multi-run simulations by analyzing ensembles of potentially complex data has become an important yet challenging visualization topic [WP09]. Key issues include providing an overview of the result space as well as relating the parameter to the result space and vice versa for sensitivity analysis and uncertainty analysis [Hel08]. If the result space consists of – or has been reduced to – one or more scalar dimensions, common multivariate visualization techniques like scatterplots and parallel coordinates can be applied [BPFG11].

Several approaches address the visual analysis of many 1D functions, typically in the context of time-series data. Visualizations may represent overlaid function graphs as envelopes [HS04], semi-transparent graphs [KMG*06], or kernel density estimates [LH11] and offer brushing techniques to highlight selected subsets of the functions [HS04, KMG*06, M*08]. Further approaches include extensions of the TableLens metaphor [KL06] and a re-orderable matrix of time series charts [MMKN08]. While an analysis of 2D function ensembles may involve 1D slices, the limitation to 1D makes these approaches insufficient for our purpose.

Potter et al. [P*09] present a framework for spatio-temporal ensembles which combines spatial summaries, trend charts, and spaghetti plots. Bruckner and Möller [BM10] split and cluster ensembles of visual effects to enable a result-driven exploration of the design space. Kehrer [Keh11] analyzes multi-run climate data using time series, glyphs summarizing statistical properties, and by brushing statistical moments in scatterplots. Nocke et al. [NFB07] also apply coordinated multiple views to analyze climate-related ensemble data. Statistical aggregations are compared in multivariate visualizations and details of single simulation runs are provided as various types of scientific visualizations. However, none of these approaches is explicitly designed for 2D function ensembles.

Based on the topological landscape metaphor, Harvey and Wang [HW10] visualize an ensemble of configurations for a contour tree as distributed icons. Busking et al. [BBP10] present a design study on the visual exploration of shape spaces and ensembles of shape models as well as approaches for a comparative visualization of shapes of 3D surfaces [BBF*11]. Our design also combines an icon-based overview of a feature-space to detailed views of an object space. Focusing on 2D function ensembles, however, we face different requirements with respect to the feature space, the information conveyed as icons, and interaction concepts. Moreover, none of these approaches addresses the problem of occlusion in icon-based overviews.

Kao et al. [KDP01] explicitly address 2D multi-run data which is treated as 2D probability distribution in a geographical context. Pixel-wise summaries show different statistical parameters for each position. Extensions of this approach cluster along spatial dimensions or group similar

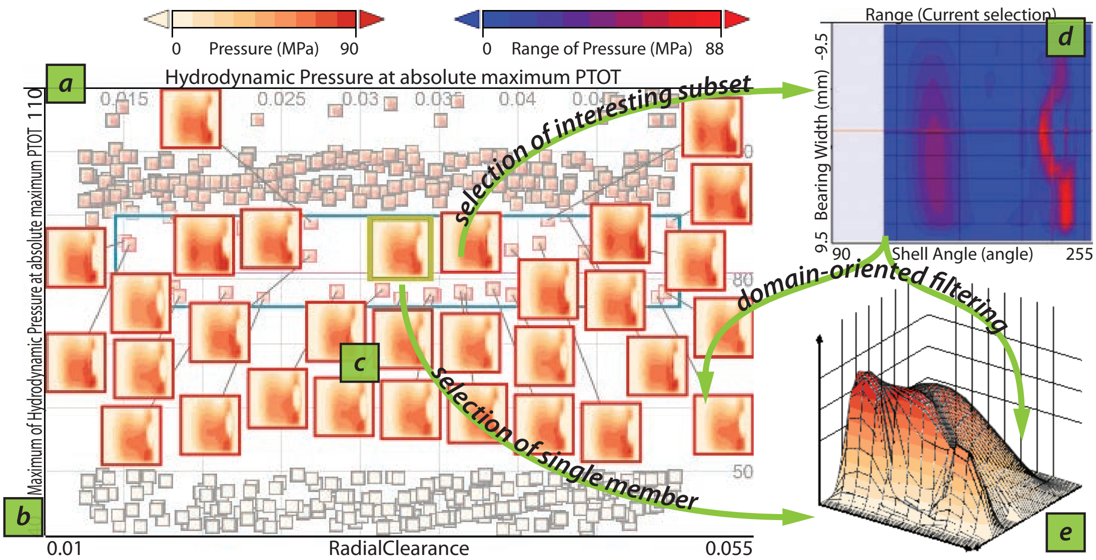

Figure 1: *Overview of the design: (a) the member-oriented overview employing feature-based placement (b) for 524 2D functions; (c) the mid-level focus of 31 selected members; (d) the domain-oriented overview showing the point-wise range of the selected subset; (e) the 3D surface plot of a single member. The arrows indicate key interactions for linking the parts.*

runs [BKS04]. Despite the usefulness as a domain-based overview, this work does not support an efficient drill-down to single 2D functions.

The work by Matkovic et al. [MGKH09] is closest to our goals. The authors describe an analysis on three levels of details, i.e., single scalar aggregates (0D), aggregated profiles (1D), and data surfaces (2D). Brushing links different levels as shown in multivariate views. However, the 0D and the 1D level lack 2D context whereas the 2D level is limited to a small number of functions. Preserving the mental model of 2D functions on all levels is a key design goal of our approach (G6). Moreover, handling densely sampled 2D functions (G7), comparing 2D surfaces (G3), and providing domain-based overviews (G1) are not discussed. We thus see a need for an approach enabling a more holistic analysis of 2D function ensembles.

3. Overview of the Design

Our design approach consists of three tightly linked parts (see Fig. 1): A member-oriented overview, a domain-oriented overview, and a 3D surface plot for details.

The *member-oriented overview* (Fig. 1a) visualizes the ensemble members (i.e., 2D functions) as icons – called *base icons*, abbreviated as "BIs" – in a 2D space. It shows distributions of member-specific features and enables member-oriented interactions (e.g., selection). Appropriate down-sampling preserves extrema even for small BIs. Options for placement (Fig. 1b) include numerous derived features and other characteristics like simulation parameters. To resolve potential overlap, we introduce a mid-level focus (Fig. 1c) which integrates larger icons – called *focus icons*, abbreviated as "FIs" – of selected members in the same visual context. Label-placement strategies avoid occlusion while minimizing the distance between corresponding BIs and FIs.

The *domain-oriented overview* (Fig. 1d) shows features across the 2D function domain as aggregated for the entire ensemble or any selected subset. It supports domain-oriented interactions which take immediate effect in the other parts (e.g., filtering in domain space or positioning 1D slices).

The *3D surface plot* (Fig. 1e) shows single members, aggregations of multiple members, or target functions. Optionally, all parts of our design enable an explicit comparison of each ensemble member to a 2D reference function and support an analysis in terms of similarity-based features.

Our implementation is integrated in a system providing various additional views for analyzing other facets of ensemble simulations (e.g., parameters, scalar and 1D results). The coordination between views is based on brushing and linking simulation runs and updating derived dimensions.

4. Member-Oriented Analysis

A member-oriented analysis treats each 2D function as individual object and comprises three levels of detail: all (Sec. 4.1), some (Sec. 4.2), and single members (Sec. 4.3).

4.1. Overview Level

The purpose of this level is to show distributions of two member-specific features for the entire ensemble in a scatterplot-like way. The ensemble members themselves are represented as rectangular base icons (BIs) with a user-defined aspect ratio (1 being the default). For placement [War02], each axis can be parameterized by different classes of features:

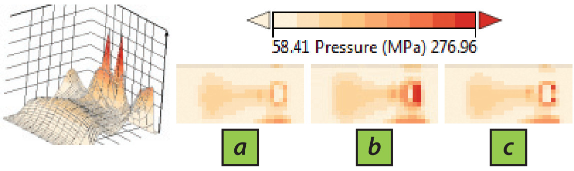

Figure 2: *Comparison of downsampling strategies on a small icon (30 x 14 pixels): (a) averaging blurs the two peaks; (b) maximum-preserving loses the gap between the peaks; (c) extrema-preserving keeps minima and maxima.*

- Statistics of the function values: Minimum, maximum, range, mean, standard deviation, 2D integral.
- The position of global extrema within the 2D domain.
- The value at a user-defined point within the 2D domain.
- Arbitrary scalar dimensions of the simulation runs (e.g., simulation parameters as on the X-axis in Fig. 1a).
- Rank-based equivalents of all parameterization options (e.g., ordering members by their maximum) to achieve a homogenous distribution.

Respective feature values are indicated by the center of each BI. Color encodes function values across the 2D domain. Based on the covered range, symmetric colormaps around zero or linear colormaps otherwise [War04] are set as defaults. The size of the BIs can be user-defined or derived from the degree of visual overlap in order to minimize occlusion. Depending on the distribution, appropriate icon sizes range from points to detailed previews. As our heuristic, automatic scaling determines the largest icon size which ensures that from all pixels of all BIs, at least 90% are visible.

All BIs – and in particular small ones – may require radical downsampling. Since extrema are the key information in our context, a downsampling strategy has to preserve them as much as possible. This forbids indiscriminate averaging. Instead, the user may – depending on the task – opt to preserve either the maximal or the minimal function values. For each pixel of a BI, all contained samples and interpolated values at its border are considered (see Fig. 2b). To minimize the loss of critical information by occlusion, this option also determines the ordering of the BIs in Z-direction.

As a third option, we offer a strategy which is inspired by topology-guided downsampling [KE01] and seeks to preserve global and visible local extrema of any type (Fig. 2c). We focus on extrema because saddle points have been described as numerically unstable in our context. The steps of the algorithm are as follows:

1. Assign all vertices of a 2D function to equally-sized blocks corresponding to the resulting pixels.
2. For each vertex: determine whether it is a visible local or global extremum. A local extremum is defined visible if it is an extremum of all vertices assigned to the own and the eight neighboring pixels.
3. Blocks with one extremum preserve that extremum.
4. Blocks containing both types of extrema prioritize global over local. For further disambiguation: preserve the one with larger absolute distance from the average of all samples (for global extrema) or the samples of the eight neighboring pixels (for local extrema).
5. All other blocks take the average of their samples.

As a nice property, the algorithm does not assume any particular sampling of the 2D functions (irregular sampling can occur in our context). The idea of visible local extrema is that a user must unambiguously determine the type of extremum based on the downsampled icon. The Z-ordering of BIs is determined by the overall extremum with larger absolute distance from the average of the entire ensemble.

Summarizing, the data-driven placement has proven to be easily understood and it flexibly generates meaningful feature distributions [War02]. Users appreciated even small BIs as immediate contextual information. We address the issue of occlusion by automatic icon scaling and to some degree by Z-ordering and rank-based placements. However, small icons are insufficient for a detailed comparison.

4.2. Mid-Level Focus

The purpose of the mid-level focus is to provide a sufficient degree of detail for interesting subsets of the ensemble. Interactions for specifying this focus level include hovering above icons, selection by rectangular brushes within the overview level, and brushes in other linked views (see Sec. 7). As design goals of the mid-level focus, the context of the overview level should be preserved whereas occlusion should be avoided. We consider distortion-oriented approaches [Hau05] a potential risk for acceptance because they are unfamiliar to engineers and could compromise the interpretability of the data-driven overview.

The main idea of our approach is to integrate large *focus icons* (FIs) for selected ensemble members in regions which are not covered by BIs of the overview level. The size of FIs is user-defined while automatic downscaling ensures that FIs never occupy more than half of the view space. The coloring is consistent with BIs. Feature-aware downsampling is also applied to FIs, albeit making use of the higher resolution. Lines connect FIs to their data-driven position.

The placement of FIs can be considered a labeling problem (see Luboschik et al. [LSC08] for a survey). Ali et al. [AHS05] describe readability, unambiguity, pleasing, real-time, frame-coherency, and compaction as general requirements of label placement. In our context, the information conveyed by a BI is also provided by its FI. We thus permit an occlusion of a BI by its FI. In sparse areas, it is desirable that FIs are positioned concentrically with their BIs to indicate the data-driven position (see Fig. 3). If a concentric placement is not possible, hovering over an FI shows its precise data-driven position. However, the placement cannot make any assumptions about the distribution of the overview level and should preserve consistency in case of incremental changes of the subset of interest.

Our approach combines a greedy algorithm for an initial placement of FIs and a subsequent force-directed step

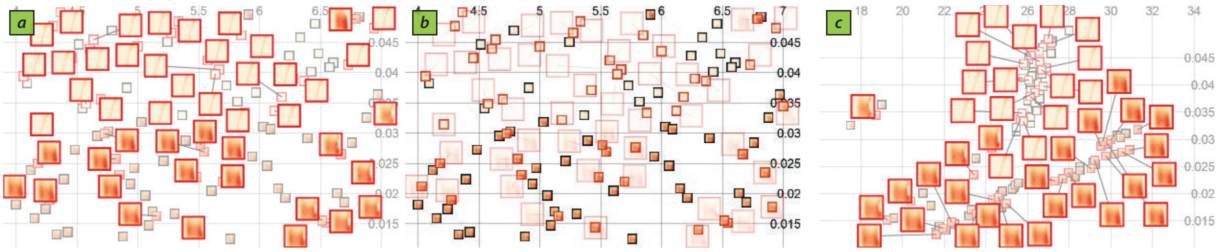

Figure 3: *The mid-level focus for different distributions: (a, b) sparse distribution where most focus icons are close to or above their base icon without occluding non-selected base icons; (c) the affiliation is still visible for densely clustered base icons.*

for a local optimization. In order to preserve the position of already placed FIs, the greedy algorithm only applies to new FIs, i.e., when the user extends a selection.We build on the labeling pipeline by Luboschik et al. [LSC08] due to its simplicity and efficiency. The pipeline describes four increasingly expensive steps (i.e., 4-position model, 8-position model, 4-slider model, distant positions using spiral sampling) and performs each step only once and only for non-placed FIs. We extend this pipeline in three ways: (1) An initial check for each FI refers to the data-driven position itself, ignoring occlusion of the own BI. (2) If FIs have not been placed at the end of the pipeline due to inevitable occlusion, we repeat the spiral sampling with an increasing tolerance with respect to occlusion. Separate thresholds for occluded pixels of BIs and FIs ensure minimal occlusion between FIs. (3) Finally, FIs are checked pair-wise for intersecting connection lines; if so, their positions are swapped. This is repeated until all overlaps are resolved or a maximal number of iterations has been reached.

Unlike the initial placement, the force-directed step affects all FIs. The idea is to achieve more fairness for subsequently selected members and to make use of previously covered space of removed FIs [AHS05]. We now describe the change in position Δ_i of a focus item i in one iteration. Let $O_i(x)$ denote the percentage of pixels of a graphical object x occluded by i. Let $C(x)$ be the center of the object x in a screen space with the width V_w and the height V_h. Let $D(x,y)$ be the normalized vector pointing from $C(x)$ to $C(y)$. Moreover, let L_f be the vector from the center of a focal item f to its data-driven position in screen space with l_f being the graphical representation of this connection line and $N_i(L_f)$ being the normalized orthogonal vector of L_f towards i.

$$\Delta_i = w_1 \frac{|L_i|}{V_w + V_h} + w_2 r_B(i) + w_3 r_F(i) + w_4 r_L(i) \quad (1)$$

$$r_B(i) = \sum_{b \in \{B \setminus b(i)\}} O_i(b) D(b,i) \quad (2)$$

$$r_F(i) = \sum_{f \in \{F \setminus i\}} O_i(f) D(f,i) \quad (3)$$

$$r_L(i) = \frac{1}{|F|} \sum_{f \in \{F \setminus i\}} O_i(l_f) N_i(L_f) \quad (4)$$

F denotes the set of FIs and B the set of BIs. $r_B(i)$ is the repulsion from BIs, $r_F(i)$ is the repulsion from FIs, $r_L(i)$ reduces the risk of an FI to be dragged across a connection line. By experiment, we identified the weights $w_1 = $ -1, $w_2 = 1$, $w_3 = 10$, $w_4 = 15$ as a reasonable choice which is insensitive to the resolution of the visualization and the sizes of FIs and BIs. The algorithm proceeds until stabilization or a maximal number of iterations. Only the final positions are shown and FIs perform a linear transition to their new position.

The mid-level focus can be considered a semantically different layer on top of the overview. The visual discrimination is enhanced by desaturating the overview layer during the presence of FIs. Hovering over an FI highlights the corresponding connection line on top of all other BIs and FIs and a dot indicates the feature-driven position in the overview level. This consistently enables a visual mapping also if connection lines are partially occluded or if an FI is placed above its BI. With regards to interaction, the main purpose of the mid-level focus is to enable an unambiguous selection of single ensemble members. However, during interactions on the overview level (e.g., feature-based selections), the mid-level focus appears semi-transparent to enable an unobstructed view on the feature space (see Fig. 3b).

4.3. Detail Level

Clicking on a focus item displays the respective 2D function as a 3D surface plot. Despite known flaws of 3D visualizations [SJOC01], adding a 3D plot was crucial for acceptance in our application domain. To preserve consistency, we apply the same coloring to function values as in the overview and mid-level focus. An additional wireframe enhances the perception of surface structures. Users may zoom and rotate the plot. A title shows the name of the simulation run.

4.4. Comparative Analysis

Comparison is a key task for most types of ensemble simulations [NFB07] and this is also true in our application domain. A comparison can take place on the image-level or data-level [PP95]. Recently, Gleicher et al. [GAW*11] categorized comparative designs as juxtaposition, superposition, and explicit encodings and they emphasized that each category has its tradeoffs. Our approach covers both image- and data-level comparison and supports juxtaposition, superposition, and explicit encodings.

The overview as described in Sec. 4.1 enables a data-level

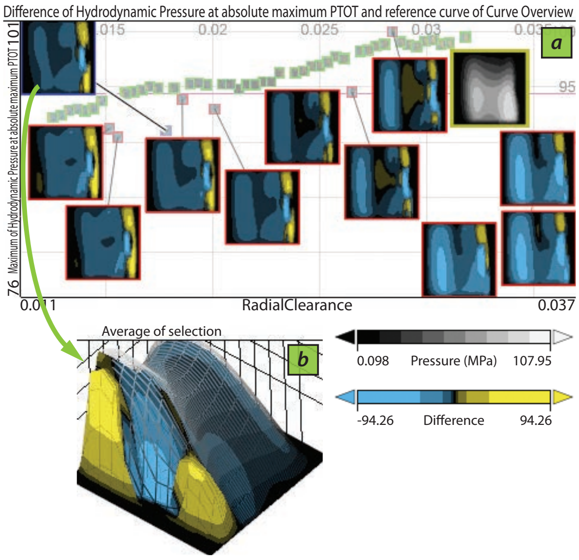

Figure 4: *The comparison mode: (a) 11 outliers are compared to a reference function (grayscale icon) being the point-wise average of a cluster; (b) detailed 3D comparison of a single member (colored surface) to the reference function (transparent grid).*

comparison via a feature-based placement and an image-level comparison by juxtaposing function icons. The mid-level focus (Sec. 4.2) enhances the image-level comparison by larger icons and avoidance of occlusion. However, an image-level comparison is rarely sufficient: Differences between ensemble members can be small and are hardly perceptible by mere juxtaposition. Moreover, a comparison should not be restricted to ensemble members and results should also be quantitative.

To address these issues, our approach supports a dedicated comparison mode. In this mode, each ensemble member is compared to a reference 2D function, denoted as $R(x,y)$, which can be a user-defined ensemble member, a point-wise aggregation of multiple or all members, as well as an external function like measured data or a target function for optimization. $R(x,y)$ is integrated as context information in all levels of detail (see Fig. 4). In the overview-level, a cross-hair indicates the feature-driven position of $R(x,y)$ if applicable. In the mid-level focus, $R(x,y)$ is added as an additional focus icon that is connected to the cross-hair (if shown), unless it coincides by definition with the focus icon of a member. The icons of all other members display the explicit point-wise differences to $R(x,y)$. Linear grayscale coloring is applied to $R(x,y)$ whereas differences are visualized using a symmetric colormap around zero (blue for negative and yellow for positive differences).

In the comparison mode, the detail level (Sec. 4.3) enables to compare one ensemble member to $R(x,y)$ by a combination of superposition and explicit encoding (see Fig. 4b). The compared member is specified by selecting a single member or by hovering over a focus icon of the mid-level focus. The respective 2D function is shown as opaque surface and color encodes the difference to $R(x,y)$ using the same symmetric colormap as the other levels. This intuitively preserves the explicit difference information in context of the 2D surface – a benefit of a 3D representation for our purpose. $R(x,y)$ itself is shown as a wireframe which applies grayscale coloring. This enables a comparison by spatial reference at least for regions like peaks or borders. Potential future extensions could include thresholds for relevance filtering [BBF*11].

To support a quantitative comparison, all types of feature-driven placement of the overview level (Sec. 4.1) can also be applied to respective difference 2D functions, i.e., the differences between each ensemble member and $R(x,y)$. This way, questions like "What is the maximal difference and for which member?", "How are peak differences distributed across the domain?" and "How do overall differences to a target function correlate to a simulation parameter?" can be answered easily. All positions are updated immediately (after a transition) when choosing a different $R(x,y)$.

5. Domain-Oriented Analysis

The domain-oriented overview (see Fig. 1d) provides an aggregated view of the 2D domain for the entire ensemble or a user-defined subset thereof, e.g., to analyze a cluster of similar members. The direction of aggregation is orthogonal to member-oriented features in that it preserves the 2D domain but aggregates across ensemble members (compare to Kao et al. [KDP01]). We support standard aggregates like maximum, minimum, range, average which are applied point-wise to the members themselves or to respective difference functions to $R(x,y)$ (see Sec. 4.4). The aggregate "count" is relevant to assess the coverage of the 2D domain by ensemble members if it is not identical for all – a rare yet possible case in our context. The coloring considers the result range of the current aggregate: The colormap of the overview is applied if the ranges of the function and the aggregate coincide (e.g., for "maximum") whereas a different colormap is enforced otherwise (e.g., for "range" and "count").

We support three types of domain-oriented interactions: (1) Rectangular filtering enables to focus on particular features in domain space. The member-oriented parts (Sec. 4) only consider the non-filtered area for icon-based visualizations and feature-based computations. (2) Defining a specific point of the domain (e.g., a peak) by a movable cross-hair enables to compare its values for all members as one type of feature supported by the member-oriented overview. (3) The position of a 1D slice can be defined along one line of the cross-hair (see the next section). All interactions trigger updates of all affected results as fast as possible.

6. 1D Curve-Oriented Analysis

An analysis in terms of 1D curves is a possible compromise between information loss and complexity. In addition to axis-orthogonal slices as defined in the domain-

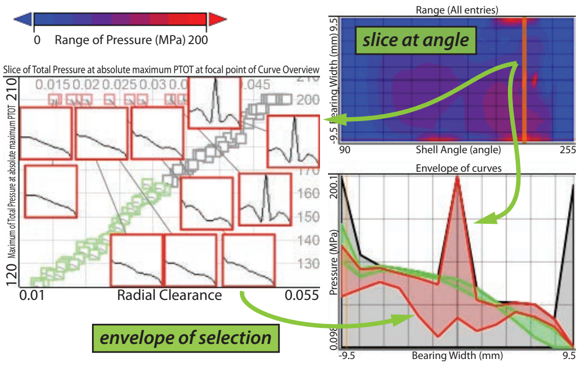

Figure 5: *Analysis of 1D slices at a user-defined position within the 2D domain. Color discriminates the envelopes of two selections.*

oriented overview (Sec. 5), we support aggregated profiles [MGKH09], i.e., aggregating one dimension of the domain while varying the other.

Our approach supports a dedicated 1D-mode (see Fig. 5) in which all types of feature-based placements can be applied to 1D data, i.e., considering only the samples along each curve. Each icon in the member-oriented overview displays the respective curve as a common function plot. Extrema-preserving sampling strategies (see Sec. 4.1) are also applied to 1D data. While color is not used in the current implementation, we intend to experiment with two-tone pseudo coloring [S*05] in the future.

In the 1D mode, the detail level displays selected curves in the context of the entire ensemble based on envelopes [HS04], i.e., the minimum and maximum of a set of curves as varying over the parameter. A gray envelope for the entire ensemble serves as context. Envelopes of interactively selected subsets are discriminated by different hues and a single black curve is shown for the reference function $R(x,y)$. In the future, an application of more advanced visualization techniques for 1D ensembles is conceivable [KMG*06, LH11, M*08].

7. Implementation and System Integration

Our approach has been integrated in a system for the visual exploration of multi-run engineering data as one type of view. Other views include general multivariate visualizations (e.g., histogram, scatterplot matrix, table lens, parallel coordinates) and approaches addressing particular tasks of our application domain [PBK10, BPFG11]. The linking between views is based on brushing simulation runs in any view, and on updating derived data attributes. For example, any feature derived from the 2D function ensemble in the proposed approach can also be assigned to any scalar attribute of any other view and is updated immediately during interaction (e.g., when filtering the 2D domain).

To enable this tight integration, all views operate on a data model that is optimized for multi-run simulation data in our application. As proposed by Konyha et al. [KMG*06] and Matkovic et al. [MGKH09], 1D and 2D functions are treated as first-level objects like scalars and strings and the API provides respective access mechanisms.

Internally, multi-threading [PTMB09] ensures application responsiveness and provides visual feedback during continuous interaction. This is essential since feature-extraction and downsampling may take a few seconds for large ensembles. All parts are written in C++ and use OpenGL for rendering.

8. Evaluation

We evaluate our approach on two levels. First, we illustrate a typical application scenario of our approach as conducted by an engineer. Second, we report user feedback collected from interviewing four experts in car engine-design.

8.1. Application Scenario

The background of the scenario is an elasto-hydrodynamic analysis of an engine bearing design [Off11]. Relevant results of a multi-run simulation include asperity contact pressure (ACP) and hydrodynamic pressure (HDP) as 2D functions of angle and bearing width. The main goal is to analyze the number and location of peaks as varying over the space of design parameters.

As a first step of the analysis, the engineer inspects a domain-oriented overview of maximal ACP for 100 simulation runs (Fig. 6a). Peaks at the upper and lower border are explained by tangential deviations. Since they are expected and less interesting, the engineer filters the analyzed domain to cover only the central peak which might indicate a potential problem. In fact, most icons in the member-oriented overview (Fig. 6b) show that central peak for cases where the parameter "force scale" (Y axis) exceeds 1. The engineer immediately explains this behavior as unplausible and hypothesizes a wrong position of the bore hole for oil supply in the engine model as the cause.

After revising the engine model, another 100 simulation runs are generated for further analysis. The engineer repeats the aforementioned steps for the new data. There are still peaks in the center, but only for 11 members. Parameterizing the Y-axis of the member-oriented overview by the maximal function value reveals an expected linear relationship between the parameter "radial clearance" and maximal ACP (Fig. 6c). However, 11 outliers are unexpected and correspond to the members with central peaks.

The engineer decides to look at HDP (i.e., another 2D simulation result) in an additional instance of our approach. Plotting "Radial Clearance" versus maximal ACP shows interesting clusters (Fig. 6d). Linking and brushing between the two instances soon reveals that these clusters in HDP correspond to groups having their maximal ACP at different positions within the 2D domain. In particular, the 11 outliers in ACP are distributed between two clusters in HDP which are thus relevant for further investigation.

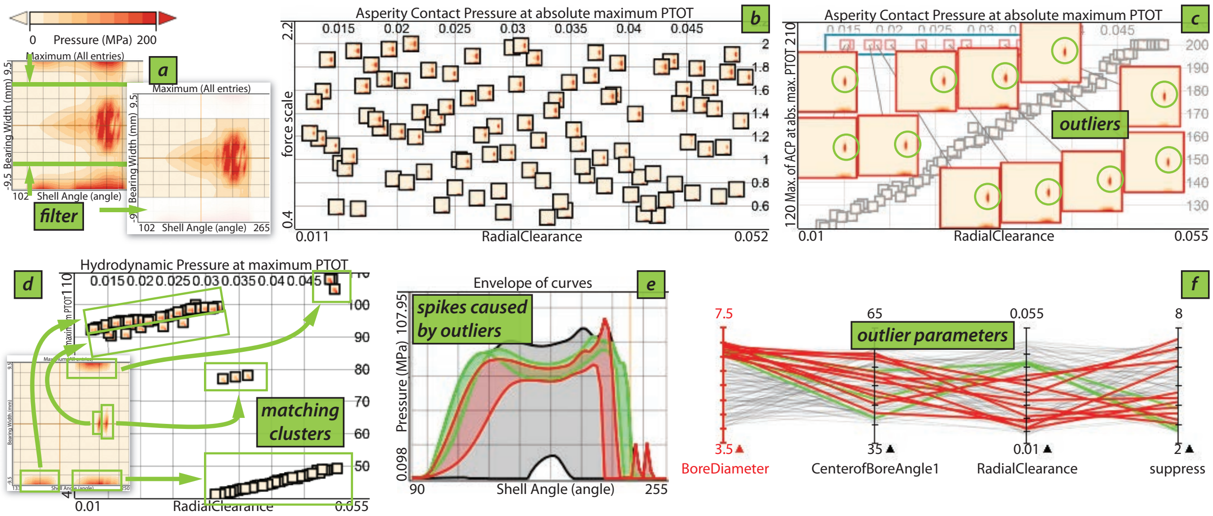

Figure 6: *An exemplary workflow: (a) filtering the domain-space focuses on a central peak; (b) most icons reveal the peak if the parameter "force scale" exceeds 1; (c) after revising the model, 11 outliers still show a central peak; (d) linking+brushing confirms that peaks in the domain-space of one 2D function ensemble correspond to clusters in the feature-space of another one; (e) slicing reveals that the outliers (red envelope) drop earlier but form additional peaks; (f) linked parallel coordinates indicate a problematic interplay of the design parameters "bore diameter" and "radial clearance".*

The upper left cluster in Fig. 6d comprises both outliers and non-outliers which make it suitable to relate shapes of outliers to cases considered normal. The engineer defines the reference function $R(x,y)$ as the point-wise average of all non-outlying members of this cluster and switches to comparison mode (Fig. 4 illustrates this step). The mid-level focus shows a similar deviation from $R(x,y)$ for all outliers in the right half of the 2D domain: HDP drops between an angle of 230 and 240 degree and a width of -3 and +3 (blue hole), surrounded by one or two peaks (yellow). Slicing at width equal to zero (Fig. 6e) confirms this finding: Outliers (red envelope) drop earlier than non-outliers (green envelope), but form additional peaks afterwards.

These peaks indicate an instable behavior which is considered an important finding. The engineer hypothesizes that the resolution of the bearing model might be insufficient for certain regions of the design parameter space. Parallel coordinates of four design parameters help to identify the regions covered by outliers (red in Fig. 6f) as an interplay of high values for bore diameter and low to moderate values for radial clearance. The green samples (non-outliers) prove that bore diameter alone does not cause the observed behavior.

8.2. User Feedback

Our target users have been assessing elasto-hydrodynamic engine bearings on a daily basis for more than ten years. Their previous workflow for this task required multiple tools and many intermediate steps (see also Sec. 1). Being restricted to aggregates was considered imprecise while comparing multiple surface plots was very cumbersome and time-consuming. The previous effort for assessing one engine bearing was estimated as three days to one week.

Due to this need, the overall feedback to our approach was very positive. The experts have been using it for three months before being asked about feedback. Using our approach, they needed on average between 0.5 and 1.5 days for a complete assessment of one engine bearing which is a time saving of approximately 70%. It was also stressed that important aspects were previously impossible at all, e.g., getting an overview of several hundred 2D functions. They agreed that our approach enables a better understanding and a higher confidence. One engineer commented, "*Where I only looked at the values of global extrema before, I can now see their location and context in the 2D domain as well*". This statement shows the importance of preserving the 2D mental model even on an overview level. Another engineer commented, "*The icon itself is the plausibility check*".

The mid-level focus was considered intuitive and efficient for an inspection of up to 50 to 100 ensemble members, depending on the monitor resolution and the data-oriented distribution. The 3D detail level was regarded essential to convey a feeling for the shape of a surface. Interactions in domain space were reported as being frequently necessary, e.g., to filter numerical instabilities at boundaries, where the visual context was described as being of much help. The experts also found the explicit comparison very valuable especially to detect small differences and to quantify them.

However, the interviewees also stressed that mastering multiple linked views took them several days for familiarization. The most critical objection concerned the suggested colormaps as they expected the rainbow colormap which is standard in their domain. They considered this critical for acceptance and we thus enabled to change the default colormap. In the future, our approach will be distributed as part of the software suite of the company AVL List GmbH. As such, it will be available to thousands of users in engine design.

9. Discussion and Future Work

User feedback indicates that our design matches the goals stated in Sec. 1.2. A comprehensive set of feature-extraction methods and a tight integration between all parts enable a flexible and holistic exploration (G1). The mid-level focus and the detail level provide an efficient access to details (G2). The comparison mode facilitates precise comparison while avoiding decontextualization (G3). Data-driven placement supports a feature-oriented, quantitative comparison (G4). Filtering and slicing enable a local analysis within the 2D function domain (G5). Icons preserve the mental model of 2D functions even on the overview level (G6). Downsampling strategies preserve extrema even for many samples and small icons (G7). Linking and brushing and derived dimensions enable a seamless integration with other views and tools of the engineers (G9).

Scalability (G8) is addressed on multiple levels. Automatic scaling minimizes occlusion in the overview level. At worst, icons become points which still show a global extremum while the scalability is comparable to a normal scatterplot. Users can selectively increase the amount of detail using the mid-level focus or slider-based zooming of the axes. Technically, employing multi-threading and dedicated graphics hardware ensures fluid interaction even for large ensembles. Our approach has been successfully tested with up to 4160 ensemble members. Larger ensembles are very rare in our domain due to a prohibitive effort for simulation.

While our approach was designed to match the needs of our application domain, we see much potential for generalization. We believe – and will investigate in the future – that our design is directly applicable to ensembles of 2D functions or 2D scalar fields from other domains. We also think that our approach could be extended to analyze image data, e.g., satellite images [KDP01] or segmentation results [TW*11]. Moreover, an analysis in terms of 2D functions can also be reasonable for ensembles of other dimensionality, e.g., slicing volumetric data.

Some lessons we have learned from our design could also be helpful when designing approaches for other types of ensemble data. The interplay between different directions of aggregation should be considered as important as defining the aggregations themselves. In this respect, we proposed interaction concepts for tightly coupling domain-oriented and member-oriented overviews. Moreover, the explicit visualization of differences (Sec. 4.4) turned out to be crucial for a precise comparison of ensemble members while juxtaposition was found to be insufficient for this task.

The mid-level focus is a novel focus+context approach for glyph-based visualizations that can be considered an alternative to distortion-oriented approaches and to relying on multiple coordinated views. As a key advantage of the mid-level focus, users do not need to shift the focus of attention to a different view for getting details which also facilitates the correspondence when focusing on multiple items simultaneously. Our target users had no difficulties in discriminating the two levels and in interpreting the different placement strategies, i.e., data-driven placement and avoidance of occlusion. However, there is a trade-off between maximizing the spatial correlation between BIs and FIs and minimizing occlusion. In our context, an FI occluding the corresponding BI was found tolerable while other configurations of occlusion are punished. Other applications may opt to avoid any kind of occlusion at the cost of less spatial correlation. As another option, the X and Y dimensions could be treated differently to largely preserve the data-driven placement also for FIs with respect to one dimension at the cost of an increased distance between FIs and BIs with respect to the other. In general, one limitation of the mid-level focus concerns its scalability to a few dozen items (see Sec. 8.2). However, we believe that the benefits potentially outweigh this limit for many applications.

We see multiple directions for future work: The design space of the mid-level focus deserves a more thorough evaluation. Context-preserving visual links [SWS*11] could address the occlusion of connection lines. Based on the concept by Berger et al. [BPFG11], a continuous analysis of a sampled parameter space could integrate a weighted interpolation of 1D and 2D ensemble members. Finally, a long-term field study will provide new insights concerning the adoption by additional groups of engineers.

10. Conclusion

This paper described a design study of an interactive visual approach to analyze 2D function ensembles in the development process of powertrain systems. The design is based on a tight integration of domain-oriented and member-oriented overviews, feature-based placement, multiple levels of detail, and support for explicit comparison. As the key challenges of our design approach, we addressed information loss by downsampling and occlusion by data-driven glyph placement. User feedback confirms a significant time saving and considerable qualitative gains for an analysis. We believe that our design may be beneficial for 2D function ensembles in other domains, and that it could serve as a basis for analyzing ensembles of images or volumes.

11. Acknowledgements

This work has been supported by the Austrian Funding Agency (FFG) within the scope of the COMET K1 program. Thanks go to AVL List GmbH for providing data and supporting the evaluation, and to E. Gröller, K. Matković, P. Muigg, and J. Kehrer for valuable comments.

References

[AHS05] ALI K., HARTMANN K., STROTHOTTE T.: Label Layout for Interactive 3D Illustrations. *J. WSCG '05 13* (2005), 1–8. 4, 5

[BBF*11] BUSKING S., BOTHA C., FERRARINI L., MILLES J., POST F.: Image-based rendering of intersecting surfaces for dynamic comparative visualization. *Vis. Comput. 27* (2011), 347–363. 2, 6

[BBP10] BUSKING S., BOTHA C., POST F.: Dynamic Multi-View Exploration of Shape Spaces. *Computer Graphics Forum 29*, 3 (2010), 973–982. 2

[BKS04] BORDOLOI U., KAO D., SHEN H.-W.: Visualization Techniques for Spatial Probability Density Function Data. *Data Science J. 3* (2004), 153–162. 3

[BM10] BRUCKNER S., MÖLLER T.: Result-Driven Exploration of Simulation Parameter Spaces for Visual Effects Design. *IEEE Trans. Vis. Comp. Graph. 16*, 6 (2010), 1468–1476. 2

[BPFG11] BERGER W., PIRINGER H., FILZMOSER P., GRÖLLER E.: Uncertainty-Aware Exploration of Continuous Parameter Spaces Using Multivariate Prediction. *Computer Graphics Forum 30*, 3 (2011), 911–920. 1, 2, 7, 9

[GAW*11] GLEICHER M., ALBERS D., WALKER R., JUSUFI I., HANSEN C., ROBERTS J.: Visual Comparison for Information Visualization. *Information Visualization 10*, 4 (2011), 289–309. 5

[Hau05] HAUSER H.: Generalizing Focus+Context Visualization. In *Scientific Visualization: The Visual Extraction of Knowledge from Data* (2005), Springer, pp. 305 – 327. 4

[Hel08] HELTON J. C.: Uncertainty and Sensitivity Analysis for Models of Complex Systems. *Comp. Methods in Transport: Verification and Validation 62* (2008), 207–228. 1, 2

[HS04] HOCHHEISER H., SHNEIDERMAN B.: Dynamic query tools for time series data sets: Timebox widgets for interactive exploration. *Information Visualization 3* (2004), 1–18. 1, 2, 7

[HW10] HARVEY W., WANG Y.: Topological Landscape Ensembles for Visualization of Scalar-Valued Functions. *Computer Graphics Forum 29*, 3 (2010), 993–1002. 2

[KDP01] KAO D., DUNGAN J., PANG A.: Visualizing 2D Probability Distributions from EOS Satellite Image-Derived Data Sets: a Case Study. In *Proc. IEEE Visualization* (2001), pp. 457–460. 2, 6, 9

[KE01] KRAUS M., ERTL T.: Topology-Guided Downsampling. In *Proc. Int. Workshop on Volume Graph.* (2001), pp. 139–147. 4

[Keh11] KEHRER J.: *Interactive Visual Analysis of Multi-faceted Scientific Data*. PhD thesis, Dept. of Informatics, Univ. of Bergen, Norway, 2011. 2

[KL06] KINCAID R., LAM H.: Line Graph Explorer: Scalable Display of Line Graphs Using Focus+Context. In *AVI '06: Proc. of the working conf. on Advanced Visual Interfaces* (2006), pp. 404–411. 2

[KMG*06] KONYHA Z., MATKOVIĆ K., GRAČANIN D., JELOVIĆ M., HAUSER H.: Interactive Visual Analysis of Families of Function Graphs. *IEEE Trans. Vis. Comp. Graph. 12*, 6 (2006), 1373–1385. 1, 2, 7

[LH11] LAMPE O., HAUSER H.: Curve Density Estimates. *Computer Graphics Forum 30*, 3 (2011), 633–642. 2, 7

[LSC08] LUBOSCHIK M., SCHUMANN H., CORDS H.: Particle-Based Labeling: Fast Point-Feature Labeling without Obscuring other Visual Features. *IEEE Trans. Vis. Comp. Graph. 14* (2008), 1237–1244. 4, 5

[M*08] MUIGG P., ET AL.: A Four-level Focus+Context Approach to Interactive Visual Analysis of Temporal Features in Large Scientific Data. *Computer Graphics Forum 27*, 3 (2008), 775–782. 1, 2, 7

[MGKH09] MATKOVIC K., GRACANIN D., KLARIN B., HAUSER H.: Interactive Visual Analysis of Complex Scientific Data as Families of Data Surfaces. *IEEE Trans. Vis. Comp. Graph. 15*, 6 (2009), 1351–1358. 1, 3, 7

[MMKN08] MCLACHLAN P., MUNZNER T., KOUTSOFIOS E., NORTH S.: LiveRAC: Interactive Visual Exploration of System Management Time-Series Data. In *Proc. Conf. Human Factors in Comp. Systems (CHI '08)* (2008), pp. 1483–1492. 2

[Mun09] MUNZNER T.: A Nested Model for Visualization Design and Validation. *IEEE Trans. Vis. Comp. Graph. 15*, 6 (2009), 921–928. 1

[NFB07] NOCKE T., FLECHSIG M., BÖHM U.: Visual exploration and evaluation of climate-related simulation data. In *Winter Simulation Conference* (2007), pp. 703–711. 1, 2, 5

[Off11] OFFNER G.: Modelling of condensed flexible bodies considering non-linear inertia effects resulting from gross motions. *Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics 255*, 3 (2011), 204–219. 1, 7

[P*09] POTTER K., ET AL.: Ensemble-Vis: A Framework for the Statistical Visualization of Ensemble Data. In *Proc. IEEE Int'l. Conf. on Data Mining Workshops* (2009), pp. 233–240. 2

[PBK10] PIRINGER H., BERGER W., KRASSER J.: HyperMoVal: Interactive Visual Validation of Regression Models for Real-Time Simulation. *Computer Graphics Forum 29*, 3 (2010), 983–992. 7

[PP95] PAGENDARM H.-G., POST F.: Comparative visualization – approaches and examples. In *Visualization in Scientific Computing* (1995), pp. 95–108. 5

[PTMB09] PIRINGER H., TOMINSKI C., MUIGG P., BERGER W.: A Multi-Threading Architecture to Support Interactive Visual Exploration. *IEEE Trans. Vis. Comp. Graph. 15*, 6 (2009), 1113–1120. 7

[S*05] SAITO T., ET AL.: Two-Tone Pseudo Coloring: Compact Visualization for One-Dimensional Data. In *Proc. IEEE Symp. on Inf. Visualization* (2005), pp. 173–180. 7

[SIBB11] SEDLMAIR M., ISENBERG P., BAUR D., BUTZ A.: Information Visualization Evaluation in Large Companies: Challenges, Experiences and Recommendations. *Information Visualization 10*, 3 (July 2011), 248–266. 1

[SJOC01] SMALLMAN H. S., JOHN M., OONK H. M., COWEN M. B.: Information availability in 2d and 3d displays. *IEEE Computuer Graphics & Applications 21* (2001), 51–57. 5

[SWS*11] STEINBERGER M., WALDNER M., STREIT M., LEX A., SCHMALSTIEG D.: Context-Preserving Visual Links. *IEEE Trans. Vis. Comp. Graph. 17*, 12 (2011), 2249–2258. 9

[TW*11] TORSNEY-WEIR T., ET AL.: Tuner: Principled Parameter Finding for Image Segmentation Algorithms Using Visual Response Surface Exploration. *IEEE Trans. Vis. Comp. Graph. 17*, 12 (2011), 1892–1901. 9

[War02] WARD M. O.: A Taxonomy of Glyph Placement Strategies for Multidimensional Data Visualization. *Information Visualization 1*, 3/4 (2002), 194–210. 3, 4

[War04] WARE C.: *Information Visualization: Perception for Design, Second Edition*. Morgan Kaufmann, 2004. 4

[WP09] WILSON A., POTTER K.: Toward Visual Analysis of Ensemble Data Sets. In *Proc. Ultrascale Visualization Workshop* (2009), pp. 48–53. 2