Vision, Modeling, and Visualization (2021)
B. Andres, M. Campen, and M. Sedlmair (Eds.)

LayoutExOmizer: Interactive Exploration and Optimization of 2D Data Layouts

Philipp Schader[1,2], Raphael Beckmann[3,5], Lukas Graner[6], and Jürgen Bernard[3,4]

[1]TU Darmstadt & [2]DKFZ Heidelberg, Germany
[3]University of Zurich, Switzerland & [4]Digital Society Initiative, Zurich, Switzerland
[5]ETH Zurich, Switzerland
[6]Fraunhofer SIT, Germany

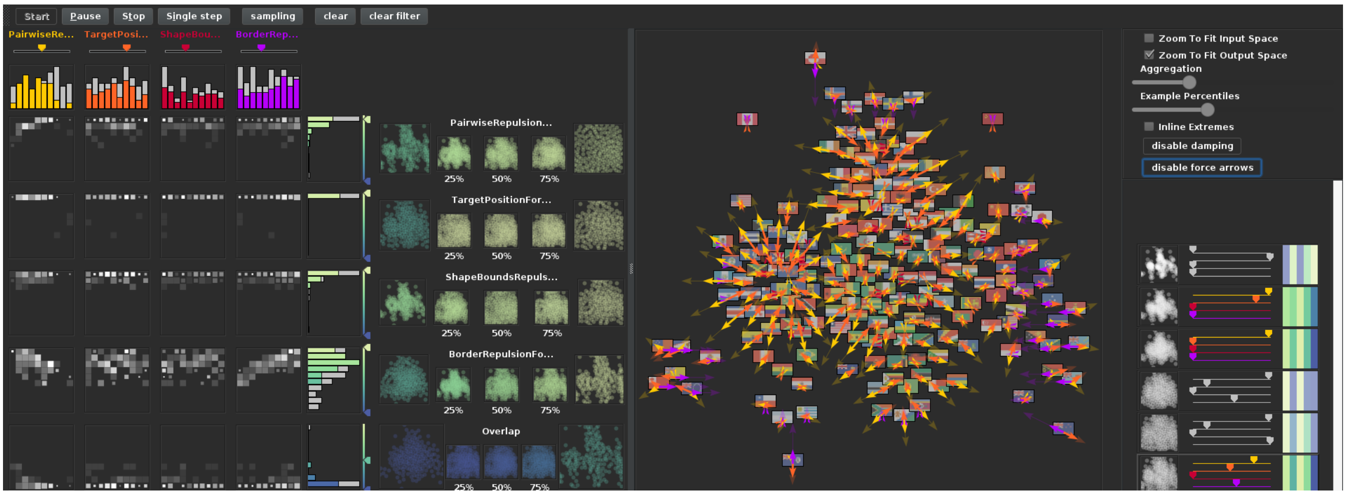

Figure 1: *The LayoutExOmizer interface allows steering a layout to optimize the positions of data points in 2D. On the left, users can refine parameters of the layout (here: four parameters in yellow, orange, red, and purple) and explore the parameter space using sampling. The visual analysis of quality measures supports users in the assessment of layouts (right part of the left view). The matrix visualization on the left also allows the analysis of correlations between input parameters and output quality measures. Interactive filtering and a history view (lower right) support the drill-down to layouts that are most meaningful for a given dataset, analysis task, and user preference.*

Abstract

Reducing the overlap of data points in 2D visualizations while preserving original positions is a challenging task. Traditionally, hand-crafted solutions have been proposed while more recently layout algorithms with a high degree of automation have been introduced. However, with a continuous parameter space, the number of alternative solutions is virtually infinite. So which one is best? This assessment can depend on many factors, coined by subjective human judgment as well as quantitative quality measures. Our approach follows the idea to have both humans and algorithms in control, to combine the strengths of both. We propose LayoutExOmizer, which stands for Layout Explorer and Optimizer. It is a visual analytics approach that guides users in finding meaningful solutions. LayoutExOmizer supports users in generating a preferred layout by discovering a corresponding set of input parameters. This parameter search is supported by visual interfaces (1) *to directly steer the parameters of the layout optimization,* (2) *to assess the quality of layouts using quality measures,* (3) *to relate input and out space, and* (4) *to filter layouts by their quality. We demonstrate the usefulness of our approach in two usage scenarios with different quality measures, including the full set of Scagnostics measures.*

1. Introduction

One of the most common visualization approaches is representing data in 2D, with position encodings in the x and y dimension of the screen space. Scatterplots are one prominent chart type, either used for numerical 2D data directly [Ans73; MF17], or for nD data in combination with dimensionality reduction (DR) [EMK*21]. For graph and network data, node-link diagrams fall into this category [VBW17], often accompanied with graph layout algorithms [CPPS20] to optimize local or global positioning. Maps directly utilize geo or spatio-temporal data [Mac04], e.g., in combina-

DOI: 10.2312/vmv.20211376

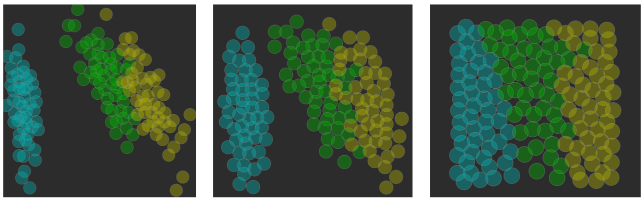

Figure 2: *Layouts for 2D data points, the original data distribution is on the left. From left to right, overplotting is removed by an increasing point repulsion force. While the layout at the center may be a meaningful layout optimization, the layout on the right is not useful as most of the original position information got lost.*

tion with projections such as the Mercator projection, mapping 3D globe data to 2D [Mon10]. Some of these techniques make use of 2D data directly; however, the majority of techniques use *layouts*, i.e., automatic methods that assemble data points in 2D. While layouts aim at reflecting structural data characteristics depending on some criterion, they also do introduce projection errors, mostly because complex data types need to be represented in only 2D. We postulate that additional local adaptions to data assemblies in 2D can be meaningful to further optimize layouts towards some optimization goal. Often, such local adaptions form an iterative and incremental layout optimization process.

One of the prevalent problems of all mentioned visualization techniques is overplotting, as soon as the number of data points exceeds the hundreds or even thousands; especially for the latter case, data points cannot be discerned as individuals anymore which hampers visualization capabilities. The problem gets worse if the point marks [Mun14] have visible area sizes: such as nodes in node-link diagrams [VBW17], marks used to show color-encoded data [JRHT14], or glyphs for data points representing inherent data characteristics [BKC*13]. The overplotting problem is aggravated if data distributions form dense regions, as is often the case when data have cluster patterns. Layout optimization is one promising line of approach to reduce local overplotting and thus increase the readability of charts. Here, the idea is to make as few modifications to the data assembly as necessary, while trying to achieve as much positive effect towards some goal (such as readability) as possible. Figure 2 shows an illustrative example of layouts being more or less useful. At a glance, a good trade-off between the preservation of positions and the reduction of overplotting is influenced by at least three different aspects. First, the involved data and its characteristics, second, the analysis task that adds requirements to layouts, and third, individual users who may deem different variations of layouts more or less useful.

To the best of our knowledge, there exists no layout method that automatically finds the optimal trade-off for any combination of these three aspects. Rather, finding meaningful layout optimizations is a human-centric approach, requiring the human in the loop. An interesting research question is what defines a good layout? At a glance, two different approaches are conceivable. First, in many research fields subjective human judgment is exploited as a meaningful source of information to answer this question. The second approach is using quality measures to assess and compare the output of algorithmic models quantitatively, aiming at covering characteristics that help to explain the goodness of layouts. From other research areas like visual cluster quality analysis [BvLBS11], class separation [AASB19], or user perception in scatterplots [WZM*19], one can learn that using (visual) quality measures or metrics [BTK11] can be useful to quantify, assess, explain, and compare the output of models. A prominent set of measures for 2D data is Scagnostics, as coined by John W. Tukey and Paul A. Tukey and later proposed by Wilkinson, Anand und Großmann [WAG06]. The line of approach pursued in this work is to incorporate both subjective human judgment and quality measures. We postulate that, with visual analytics techniques, users may be enabled to effectively define trade-offs in data layouts, leading to 2D data representations that are particularly useful for a given dataset, analysis task, and user preference. We put emphasis on cases where point marks of several hundreds of data objects have visible area sizes, as it is the case for icons, images, thumbnails, or glyph designs. Targeted user groups include visualization designers with the goal to present information, as well as experts using interactive data science tools, both requiring layout optimization in an interactive and incremental process.

Our primary contribution is LayoutExOmizer, a visual analytics tool that allows the steering, control, analysis, and assessment of 2D data layouts. LayoutExOmizer uses a real-time layout optimization model based on forces that can be steered by users through parameters. Visual quality assessment supports the analysis of optimization effects, e.g., by incorporating the nine Scagnostics measures. Visual comparison techniques ease the comparison of data layouts across hundreds of alternatives, either created manually or by the automatic sampling of layout parameters. Interactive overview and detail as well as filtering interaction support the effective identification of layouts that are most useful for a given dataset, focused analysis task, and user preference. We provide evidence for the effectiveness and efficiency of LayoutExOmizer through two usage scenarios, one with five quality measures applied on a geo dataset showing the positions of countries' capitals on Earth, and one with the nine Scagnostics quality measures applied on the dimensionality-reduced Iris dataset. With the support for quality measures, this work represents one step towards the design of fully automatic layout optimization methods that will be able to take data characteristics, analysis tasks, and user preferences into account.

2. Related Work

We structure related works by algorithmic models that achieve 2D data layouts, approaches supporting users in the exploration of parameter spaces, and approaches for visual quality assessment.

2.1. Layout Optimization

Layout optimization is a challenge that many communities face, be it removing overlaps in geo-referenced data [vGPNB17], creating direction-preserving layouts [SBMK14], graph-based and node-link layouts [CPPS20], or optimizing the placement of labels within scatterplots [MvGBW19]. Also, DR [EMK*21] relates to layout optimization, mapping high-dimensional data in low-dimensional spaces. Especially non-linear variants [VPN*10] often utilize local layout optimization criteria, such as neighborhood preservation [VK06], stress minimization [Kru64], or cluster preservation [vdMH08].

To structure layout optimization algorithms for 2D data, we borrow the notion of *control* [Shn20] in the layout process, which is either on the human side (human involvement), on the machine side (automation), or both (mixed initiative, semantic interaction, and visual analytics). On one side, entirely hand-crafted layouts exist, such as the London Metro Map, which was created in the 1930s by Henry Beck [GM94]. While this approach yields a lot of flexibility, it misses out on the potential of computer-based optimization. On the opposite side of the spectrum, there are fully automated approaches, designed to return an optimal solution given a set of constraints. One example class of methods are graph optimization algorithms, which can, e.g., be force-directed [FR91] or based on stress majorization [GKN05]. Layout optimization problems are known to be NP-Complete [HIMF98], which is why many heuristics and greedy algorithms were designed [Lyo96]. For example, Micallef et al. proposed a model with a carefully fine-tuned loss function, which, given some high-level user objectives, automatically finds optimal parameters for the scatterplot [MPOW17]. Those approaches are efficient and deliver reasonable results but lack any form of user input and are thus unable to support user preferences and iterative refinement. Parameters are a natural way of how both humans and models can have control in the layout optimization process, resulting in alternative solutions depending on the defined parameter values. Example methods are t-SNE [vdMH08] as a representative for DR or the LinLog energy model [Noa03] for graph layouts. It is up to the user to identify and classify changes in the output layout, and how to refine the parameters to get to better solutions. Our approach follows the visual analytics principles [SSS*14] where users can steer layout parameters, calculate alternative solutions, analyze the layout quality, and compare it to other solutions. To the best of our knowledge, no visual analytics approach exists that supports users in the layout optimization process including both human control and a strong degree of automation. Inspiration comes from related approaches focusing on other application goals, such as multi-criteria decision-making [PST*17], PCA-based DR [JZF*09], interaction in the DR context [SZS*17], the interactive generation of 1D layouts [NL20], or algorithms that automatically generate optimal layouts for a scatterplot [MPOW17]. The main differentiator between the latter and our approach is the native support for different shapes and glyphs, and the addition of human control and feedback.

2.2. Parameter Space Analysis

Our approach relates to parameter space analysis, as we share several common problems and analysis tasks: given a controllable input for a model, which effect will some input have on the output? In turn, presuming that a preferable output is measurable, which input parameter values lead to such a desirable output? How does the space of input parameters relate to the distribution of measures calculated on the model output? A conceptual framework for visual parameter space analysis was proposed by Sedlmair et al. [SHB*14]; we use the framework to describe related work according to three conceptual dimensions: data flow model, navigation strategies, and analysis tasks. First, the data flow model depicts how data is generated and manipulated in a visual parameter space analysis setting, which is based on sampling in our case. Examples include stratified random sampling, systematic random sampling, and uniform sampling [APSN13] (as in our approach). Second, navigation strategies describe how data was made available for navigation. Our approach supports an informed trial-and-error strategy, as users are enabled to interactively run a layout optimization model with a specific parameter setting to create one new sample, inspect the output, and re-run the model with refined parameter values. Besides, our approach mainly follows the so-called global-to-local strategy, which is in line with Shneiderman's information seeking mantra: gain an overview of all pre-computed samples and then drill-down into more details, by using filtering operations in our case. Related visual analytics approaches include Bruckner and Möller's approach to support visual effect designers in finding desired explosion animations [BM10], the visual parameter space analysis approach for image segmentation proposed by Torsney-Weir et al. [TSM*11], the WeightLifter approach in the context of multi-criteria decision making by Pajer et al. [PST*17], or the Vismon design study on fisheries management by Mooshehrian et al. [BMPM12]. The third dimension is understanding the tasks that users engage in when doing visual parameter space analysis, with the set of fine-grained tasks optimization, partitioning, fitting, outliers, uncertainty, and sensitivity. Our approach aligns well with optimization, referring to finding the best parameter combination given some objectives [SHB*14]. Optimization tasks often require the ability of subjective human judgment [BM10], numerical quality measures [TSM*11], or both, as in our approach.

2.3. Measures and Metrics

Measures and metrics are used in a wide range of applications to help to extract meaningful information from complex and high-dimensional data and to provide decision-making support [BTK11]. Our research falls into the category of measuring visualization characteristics and layout quality. An early example of such metrics is Tufte's data to ink ratio [TG83]. Other research aims at quantifying the goodness of DR techniques, e.g., based on the quantitative assessment of neighborhood preservation [LV10], or on rank-based quality criteria [LV09]. As for layout optimization, multiple criteria exist that can be measured. Recently, work has been done to understand human judgments in scatterplots, e.g., in connection with class separation measures [SA15]. Similarly, our approach provides a first step in this direction for layouts, as it brings together layout optimization, quality measures, and visual interfaces for human judgment. Arguably the most prominent set of measures specifically for scatterplots are the nine Scagnostics measures, following the idea to describe visual features in 2D data distributions formally. Our approach uses Scagnostics as a default set of measures for the assessment of layout quality.

3. Abstractions and Non-Visual Support

3.1. Data Abstraction

Our approach is applicable to datasets where data points are represented with two numerical attributes. In turn, these two attributes form the basis for the visual mapping to the (initial) display to positions. Different input data and application scenarios are possible:

- 2D numerical data (leading to a scatterplot)
- Multivariate data in combination with some DR method (leading to a scatterplot or similar)
- Graph or network data in combination with some graph layout algorithm (leading to a node-link diagram)
- Geo locations attributed with latitude and longitude information, possibly in combination with a projection such as the Mercator projection (leading to a map)

According to our design target of showing points with area sizes, a recommendation is to have a glyph, icon, or image available for every data point, as, e.g., demonstrated in Usage Scenario 1.

3.2. Task Analysis

The goal of our approach is to enable users to optimize 2D data layouts for up to several hundred data points with area sizes. For the design of a visual analytics approach, we subdivide this goal into actionable tasks, following a global-to-local strategy [SHB*14]:

from the creation, exploration, quality assessment, and filtering of multiples to the inspection of singles and preserving provenance.

T_1: Steer the layout interactively. By steering the input parameters of the data layout method interactively, users gain control over the layout. It may be useful to provide semantically interpretable parameters, such as *pairwise repulsion*, or *preservation of original position*. This fosters the informed and target-oriented parameter steering and reduces ill-defined try-and-error operations.

T_2: Explore layout parameter space. When layouts contain several (continuous) steering parameters, the number of different layouts is infinite. It is necessary to support the exploration of the parameter space to learn about parameter characteristics and their interconnections, as well as to form a basis for the identification of local and global optima. Along these lines, it may also be beneficial to provide sampling routines to cover the parameter search space.

T_3: Assess and compare layout qualities. Users may want to assess the quality of layouts in different ways. First, it is desirable to show the current layout in detail, so that users can judge the layout by subjective criteria. Second, quality measures for data layouts may be provided, allowing the system to give quantitative feedback about layout characteristics and layout quality. Finally, given large numbers of possible layouts (either steered by hand or crafted by sampling routines), users need a means to compare these layouts with respect to quality differences. The comparison method also serves as a component to guide users towards meaningful layouts.

T_4: Relate parameter space and quality measures. A classical task borrowed from (visual) parameter space analysis is to foster input-output analysis, i.e., to seek relations between the input and the output space. Depending on the characteristics of input parameters and output measures, these relations are not necessarily linear, leading to the problem that by using inappropriate correlation analysis support, some interesting relations may be overlooked.

T_5: Filter layouts. To drill-down to few meaningful layouts from a large set of candidates, users need filtering capabilities. This type of functionality is inspired by dynamic query approaches. A technical challenge is to unify filtering across different quality measures in a joint filter model. A design challenge is to support both the analysis of all samples and those that are currently filtered in.

T_6: Preserve layout analysis history. Users may rely on the history of layouts that they have created, identified, or analyzed in detail. Special requirements for this type of provenance information may be three-fold: the representation of the layout itself, the input parameter values, and the output quality for every measure.

3.3. Optimization of 2D Layouts

While plotting data points in 2D is a common approach for displaying the interrelation between the dimensions, this approach often leads to overplotting where individual data points obscure others. To mitigate the dilemma between displaying the true position of a data point and the introduction of a displacement to reduce overplotting, layout optimization methods can help, e.g., based on the interpretation of a 2D layout as a mass-spring system. The forces which are applied to the individual data points represent aspects of the layout and can be weighed for further emphasis.

The LayoutAnalyzer † is a tool allowing the optimization of 2D Layouts via the mass-spring approach. It is the layout optimization algorithm we choose for the LayoutExOmizer. It introduces 4 distinct forces to represent different aspects of a layout: The **Pairwise Repulsion Force** introduces pairwise repulsion between data points unaffected by the overlap of their shapes. The **Target Position Force** attracts data points to their true position. The **Shape Bounds Repulsion Force** introduces pairwise repulsion between data points if their shapes are overlapping. It addresses the overplotting problem directly. The **Border Repulsion Force** introduces a force towards the center of the data point cloud proportional to the distance of a data point to the bounding box of the data point cloud. This leads to more compact shapes.

3.4. Layout Quality Measures

To quantify the characteristics of 2D layouts quality measures are needed. A popular choice are the *Scagnostics* measures. Additionally, we derived quality measures directly from the mass-spring model itself. The length of the force vectors

for a specific force can be used as a measure of how well a force can be minimized and thus describe how well the aspect represented by this force is present in the resulting layout. Using this principle, four quality measures can directly be derived from the available forces: **Pairwise Repulsion Force Length**, **Target Position Force Length**, **Shape Bounds Repulsion Force Length**, and **Border Repulsion Force Length**. In addition, we establish an **Overlap** measure, defined as the percentage of shapes in the layout that have at least one intersection to other shapes. This measure allows a simple quantification of overplotting in a layout.

4. LayoutExOmizer Interface

We present LayoutExOmizer, a visual analytics system that enables users to optimize 2D data layouts interactively. With LayoutExOmizer, users can steer layout parameters, each forming a force in a mass-spring layout (see Section 3.3). Multiple quality measures support users in the assessment of layout characteristics and qualities. Finally, users can drill down to meaningful layouts by using filtering controls and a layout history. The LayoutExOmizer interface is shown in Figure 1. In Section 4.1, we provide an overview of the system, before we describe how LayoutExOmizer addresses six abstracted analysis tasks (see Sections 4.2 to 4.7).

4.1. LayoutExOmizer Overview

We designed LayoutExOmizer with a black background so that especially bright highlight colors stand out. This is particularly useful as we make use of two colormaps, each sharing half of the hue circle. One colormap is used to encode individual layout parameters, by default using categorical colors (yellow, orange, red, and purple). These colors are also used to link parameter visualizations across different views. The second colormap is unipolar with colors from dark blue to bright yellow and encodes the value domains of individual quality measures. LayoutExOmizer is a visual analytics approach with individual layouts being the primary data object: all interactive selection and filtering capabilities are fed back to a selection and filter model working with layouts; the effects are also linked across views.

The current layout state is always visualized at the center, with the option to show color-encoded forces in detail. At the top left, controls are provided to support individual analysis tasks. Also on the left are the views to support steering the layout (T_1), explore the

† https://github.com/javagl/LayoutAnalyzer

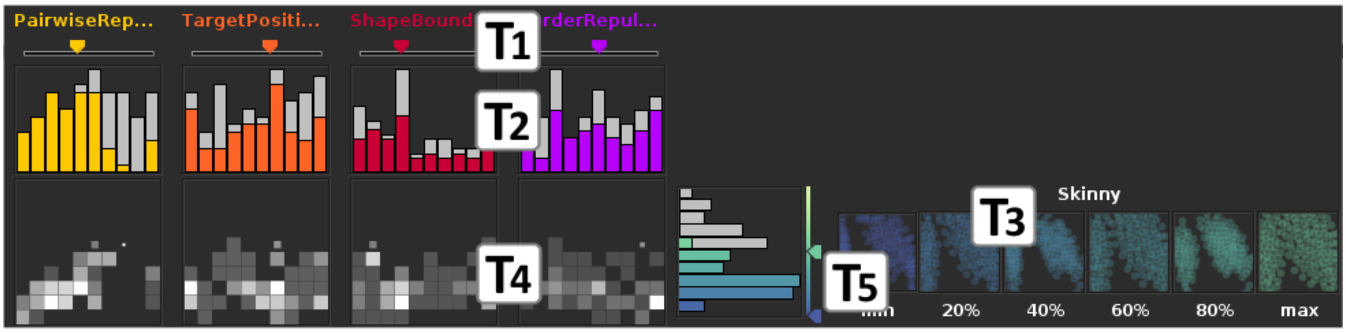

Figure 3: *Detailed overview of the visual interfaces supporting* **T**$_1$ *to* **T**$_5$*. Users can steer parameters* **T**$_1$*, explore parameter distributions* **T**$_2$*, assess the quality of layouts* **T**$_3$*, related input and output* **T**$_4$*, as well as filter layouts by quality* **T**$_5$*.*

parameter space (**T**$_2$), assess layout qualities (**T**$_3$), relate the input and output (**T**$_4$) and filter by quality (**T**$_5$), as shown with task labels in Figure 3. On the right, additional configuration parameters share the display space with the history (**T**$_6$) view at the bottom.

4.2. T$_1$: Steer the Layout Interactively

LayoutExOmizer offers a start button to start the layout optimization model and buttons (pause, stop, single step) for further layout control. Users can steer the layout and observe the converging result in real-time in the layout view at the center. This forms the basis for an interactive, iterative, and incremental layout optimization process. To steer and analyze layouts, LayoutExOmizer offers sliders to adjust the four color-coded parameters. In the controls view on the right, the visualization of force arrows can be enabled for the large layout view at the center, showing how the four parameterized forces pull each data point in the layout. The length of these arrows correlates with the magnitude of the forces, and the color coding is identical to one of the respective parameters.

4.3. T$_2$: Explore Layout Parameter Space

To populate the parameter space exploration with data, LayoutExOmizer samples values uniformly from the parameter space (triggered by the sampling button), the number of samples is a user parameter. A high number of samples further supports downstream tasks, including the layout quality assessment (**T**$_3$) and the analysis of input-output relations (**T**$_4$). Histograms in the same colors as the parameters support the analysis of parameter value distributions, as it can, e.g., be seen in Figure 3 in detail for the four parameters.

4.4. T$_3$: Assess and Compare Layout Qualities

The design of LayoutExOmizer provides two complementing solutions for the assessment and comparison of quality measures. Both

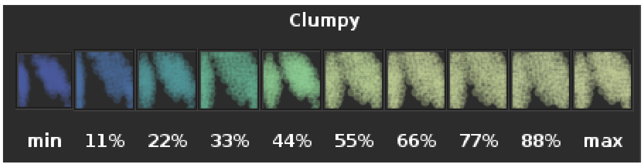

Figure 4: *The percentile examples view shows the behavior of quality measures (***T**$_3$*). Representative layouts for different percentiles of measure values are shown. The number of representatives is a user parameter. Here, the Clumpy measure is shown for ten layouts, revealing a nice semantically-interpretable order.*

share a common colormap with dark blue values for low qualities and bright yellow values for high qualities. The first interface helps users to make sense of quality measures and to understand their behaviors. This is especially useful if quality measures are unknown in the beginning, or require some sort of validation and trust-building. The idea is to show a small but representative number of layouts to the user, ordered and selected by certain percentiles of the sampling distribution of quality measure values. By default, five layouts are shown (min, 25%, 50%, 75%, and max percentile), as, e.g., shown in Figure 1, forming a semantically interpretable order of layouts across the value domain of the quality measures. The number of representatives is a user parameter: in Figure 4 ten representatives are shown. A click on a percentile preview icon automatically selects the particular layout by refining the input parameters, accordingly. In the following, we refer to this interface as the *percentile examples view*. The second interface uses histograms to show the value distributions for every measure. This is useful when it is unclear if layouts for a given dataset lead to results that are feasible for a quality measure. Heavily skewed distributions as shown in Figure 1 (second and third measure) are examples for measures that are not useful for a given application context.

4.5. T$_4$: Relate Parameter Space and Quality Measures

To understand the interactions between input parameter space and output quality measures, LayoutExOmizer offers heatmaps for every cross-cut between parameter and quality measure, leading to a grid-based visualization in the notion of an aggregated scatterplot matrix. In every heatmap, white stands for a high number of input-output combinations, whereas black stands for no occurrences. With the heatmap, users can identify relations and the type of relation, such as a positive linear correlation or a negative non-linear correlation (both can be observed in the two usage scenarios). The main advantage of heatmaps over standard scatterplots is its visual scalability, as it does not suffer from overplotting problems for many samples. Figure 5 shows an example heatmap with a negative non-linear correlation. The figure also demonstrates how the interface changes with users change the aggregation level from coarse (left) to fine-grained (right). In general, the awareness for relations helps users to understand what consequences a change in the parameter space would presumably have in the quality measures, and vice versa. As such, the heatmap works as a guidance component to make informed decisions for parameter steering.

4.6. T$_5$: Filter Layouts

LayoutExOmizer offers filtering capabilities for every quality measure. This enables users to exclude layouts that do not match given

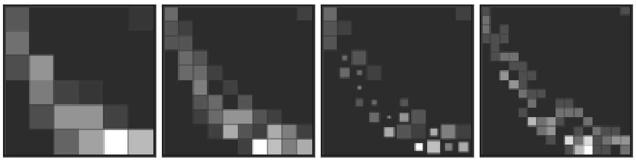

Figure 5: *The aggregation level in the histograms for input-output relation seeking (**T**$_4$) is a user parameter. The third example also shows how histograms respond to filtering operations.*

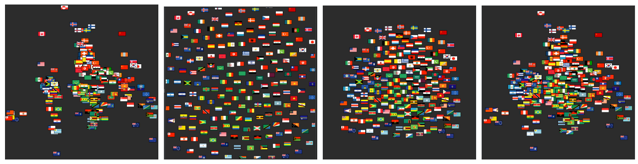

Figure 7: *Four different layouts of the countries dataset. Original layout (left), too much pairwise repulsion force (second from left), too much border repulsion force (third), and the final layout (right).*

quality criteria. Filtering is implemented with a range slider, in the notion of the dynamic queries principle, and is aligned vertically with the histogram for quality measure values, as provided for **T**$_3$. Samples that are filtered out are automatically grayed out in all histograms, both for parameter distribution charts and the quality measures. The relation-seeking heatmaps for **T**$_4$ also respond to filtering: a size encoding represents the filter state for every individual cell (see Figure 5 right).

4.7. T$_6$: Preserve Layout Analysis History

The view at the bottom right is designed to compare recent layouts in a history (**T**$_6$) visualization, with the most recent one being at the bottom. The history elements can be used to undo operations and to recover a certain parameter setting if the user decides to do so. An enlarged version of the interface is shown in Figure 6. The left part shows a preview of the actual layout, the corresponding parameter settings are visualized in the middle. On the right, a color coding shows the values of all involved quality measures. If a history item is filtered out, it will be colored in gray such that users can distinguish history items with respect to the filter status.

5. Evaluation

5.1. Usage Scenario 1

In this scenario, we optimize a map visualization with capitals of 193 countries[‡], the original 2D positions of these capitals are based on a Mercator projection. From a user's perspective, we represent every capital by its country flag with point marks of reasonable area sizes so that flags are well recognizable, and we form the requirement that flags of almost every country should be discernible in the final layout. Data-centric constraints can be seen in the initial layout in Figure 7 (left): dense regions on Earth such as Europe yield massive overplotting problems, whereas other regions remain empty. From a task perspective, the resulting layout should be able to separate continents, as well as preserve local neighborhood relations between countries. We will identify a layout optimization that considers the three aspects users, data, and task as shown in Figure 7 (right). We use the layout optimization model in combination with the set of five Layout Quality Measures (see Section 3.4).

At start, we steer the layout (**T**$_1$) manually to better understand the behavior of the layout optimizer. The second layout in Figure 7 is the result of a very high *Pairwise Repulsion Force*, leading to an overplotting-free representation which, however, does not provide any structural characteristics of the data positions any more; some point marks even left the visible display space. To encounter the latter, we use the *Border Repulsion Force* to push outlying points back towards the center: an unused area at the display borders emerges (third layout in Figure 7). What can be inferred from only a few examples is the existence of numerous layouts with serious shortcomings that can be created with the four-dimensional parameter space, i.e., there is multiple ways to fail in the layout creation.

To explore the parameter space further (**T**$_2$), we apply sampling and create 200 additional layout optimizations, the value distributions of the four parameters can be seen in Figure 8. Along these lines, we also analyze the five quality measures (**T**$_3$) and make some interesting findings: by looking at the value distributions in the quality measure histograms, we infer that the PairwiseRepulsion and the BorderRepulsion measures are descriptive across the 200 layouts, whereas the distributions of the TargetPositionForce and the ShapeBoundsRepulsion do not seem to be useful to characterize or discriminate layouts for the given dataset. The input-output analysis (**T**$_4$) reveals strong correlations of the Pairwise Repulsion Force with most quality measures. Interestingly, three of these relations appear to be non-linear. The interpretation of these correlations reveals that, e.g., high Pairwise Repulsion Force reduces the overlap of points significantly, which is to be expected from a semantic point of view. However, too much pairwise repulsion leads to layouts with low preservation of original point coordinates as, e.g., the second example in Figure 7 shows. We use the filtering (**T**$_5$) controls to exclude layouts suffering from the discussed drawbacks. We filter out layouts with particularly high data point overlap but also layouts with particularly high pairwise repulsion. The result can be seen in Figure 1. For the Pairwise Repulsion Force (yellow input parameter) layouts with extreme parameter values have been removed thanks to the filtering operation, which gives the advice to avoid particularly low and high parameter values to receive useful layouts.

Informed by these findings, we further refine the parameter values. We also consider a semantic criterion for the strength of the Target Position Force: we increase the parameter in a way that gaps between continents remain visible. The history of manually steered layouts (**T**$_6$) can be seen at the bottom right of Figure 1, including our final layout candidate that is also shown in Figure 7 (right). We have arrived at a meaningful trade-off between overplotting mitigation, preservation of original coordinates, and readability of flags.

‡ http://archive.ics.uci.edu/ml/datasets/Flags

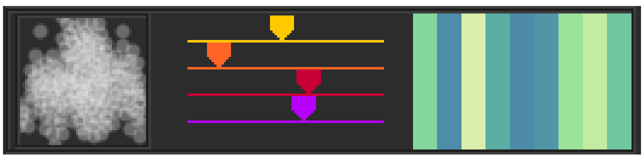

Figure 6: *History element showing a layout preview on the left, the color-coded parameter values at the center, and a fingerprint of quality measures (here: Scagnostics) on the right.*

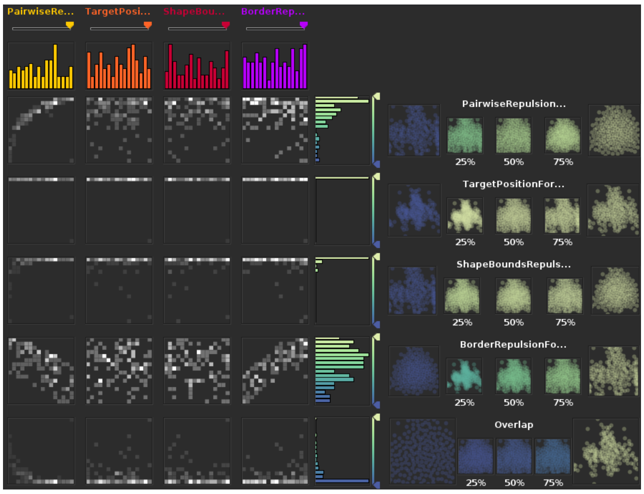

Figure 8: *Sampling of 200 layouts with various parameter values and value combinations. The parameter space analysis (***T**$_2$*), assessment of quality measures (***T**$_3$*), identification of input-output correlations (***T**$_4$*), and filtering of layouts with meaningful qualities (***T**$_5$*) can be facilitated with the combination of shown views.*

5.2. Usage Scenario 2

We use the frequently applied Iris[§] flower dataset to demonstrate the usefulness of LayoutExOmizer in the context of DR. Data with more than two dimensions can be mapped into 2D in an upstream process, followed by a layout optimization step with LayoutExOmizer. The 4D iris dataset contains three classes of iris plants (50 data points each), all mapped into 2D using non-linear MDS [Kru64]. From the user's perspective, we want to represent each data point with a point mark of a considerable size, so that color can be used to differentiate flowers by their three classes. The data characteristics reveal strong overplotting in the center regions of all three classes, as can be seen in the left view in Figure 2. From a task-based perspective, it is our goal to enable the analysis of class separation of every individual data point. To account for these three aspects, we use the layout optimization model as introduced in Section 3.3. In contrast to Usage Scenario 1, we demonstrate the approach with the Scagnostics quality measures (see Section 3.4). The supplemental material shows a system overview screenshot at large, showing the four forces (left), nine Scagnostics measures (left), the current layout with forces (center), and five history points (bottom right) at a glance. A cutout version is shown in Figure 9, with interfaces for $\mathbf{T}_1$ to $\mathbf{T}_5$.

We start with steering the layout ($\mathbf{T}_1$) manually, to better understand the effect of forces on the dataset. One example of weak layout quality can be seen on the right of Figure 2, where a high Pairwise Repulsion Force, a high Shape Bounds Repulsion Force, and a high Border Repulsion Force led to a layout with weak neighborship preservation (judged visually) and large empty image border spaces (a figure with visible forces of the layout is included in the supplemental material). To start the parameter space exploration ($\mathbf{T}_2$), we apply sampling and yield 100 alternative layouts, the result can be seen in Figure 9. We are interested in the Scagnostics measures and focus on the interface for the visual quality assessment ($\mathbf{T}_3$): the interpretation of the nine measures reveals quite

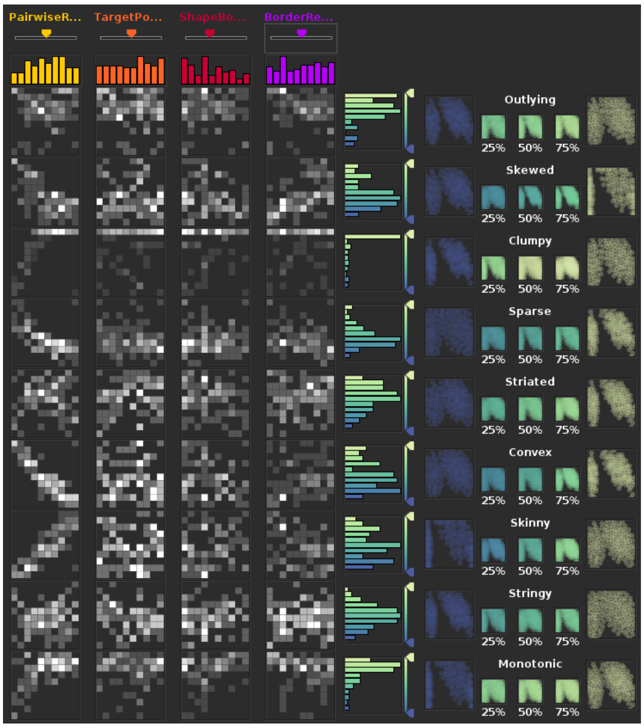

Figure 9: *Visual analysis of 100 layouts of the iris dataset. Cutout version of a system screenshot showing the interfaces for* **T**$_1$ *to* **T**$_5$*. At a glance, we identify different behaviors of the nine Scagnostics quality measures (histograms) as well as intersting correlations between parameters and qualities (heatmaps in the grid).*

different findings. By looking at the quality histograms, we identify that the Clumpy measure is not useful for this dataset: most layouts yield very high values, thus, the measure hardly helps to discriminate layouts. In contrast, the measures Skewed, Striated, Skinny, and Stringy are particularly able to discriminate layouts, as the individual layout quality values almost form normal distributions. As a next step, we want to analyze these measures in detail to gain a better understanding of their behavior. We use the percentile examples view to assess the behavior of quality measures, as shown in Figure 4. Clumpy shows a nice ordered distribution from compact data points (left) to distorted data points (right). We do the same for other measures and observe that Convex seems to measure an inverse behavior of Clumpy. The next step includes relation-seeking between input parameters and output qualities ($\mathbf{T}_4$), again facilitated with Figure 9. Some of the observations include: The yellow Pairwise Repulsion Force has the strongest relations to quality measures across all forces. In particular, we identify a positive correlation with the Skinny measure and a negative correlation to the Convex measure. From an output perspective, we identify measures that hardly show any correlations with input parameters such as the Stringy measure. Such measures may still be useful for layout filtering, but do not provide direct guidance for parameter steering.

We drill down the number of layout candidates to arrive at a final layout. We filter ($\mathbf{T}_5$) lowest Sparse values, as these layouts turn out to be too distorted (visual judgment using the percentile examples view $\mathbf{T}_3$). Similarly, we filter out high Clumpy and high Skinny layouts. The result can be seen in Figure 3, where we only focus on the Skinny quality measure (for illustration purposes). With the current filter status, the yellow Pairwise Repulsion Force on the left

[§] https://archive.ics.uci.edu/ml/datasets/iris significantly got skewed towards the left: this is a clear indication to use this steering parameter with care and assign a rather low value, as too much pairwise repulsion leads to distorted layouts. Along these lines, we refine all four parameters, and arrive at a final layout, as can be seen at the center of Figure 2.

6. Discussion and Future Work

6.1. Alternative Quality Measures

The Scagnostics measures and our Layout Diagnostics measures are beneficial to quantify and assess characteristics of 2D data distributions as well as of the layout. Other measures may also be useful, depending on the dataset, analysis goal, and user preference. An alternative is the size of the (glyph) mark, in cases when users aim at maximizing this visible size. A final example points towards classification tasks with labeled (colored) data points. Here class separation measures may be particularly beneficial.

6.2. Generalization of "Goodness" of Layouts

Layout optimization encompasses different, sometimes even contradicting optimization goals. We argue that finding trade-offs can be done effectively with human-in-the-loop approaches. To support users further, we use quality measures to quantify optimization goals formally. By doing this, our work constitutes one step towards the design of fully automatic layout optimization methods, taking complex and human-centered optimization goals into account. Future work should include empirical user studies to formalize favorable and unfavorable layout characteristics in scatterplots.

6.3. Scalability

The scalability of the layout optimization depends on the number of forces and their complexity as well as the amount of data points. If there are n data points, k forces and it is presumed that every force has a time complexity of $\mathcal{O}(n^2)$ in the worst case, the total time complexity of one simulation step is in $\mathcal{O}(kn^2)$.

The frame rate on an Intel Core i5-9400F powered workstation was evaluated using artificial data points sampled form a 2D normal distribution. For data set sizes of $n = 100, 300, 600, 900$ the average frame rates are $79.74, 10.73, 2.64$ and 1.02 respectively Visual scalability is limited by the two factors parameter count and quality measure count, both of which should not exceed ten to preserve all data analysis capability at one screen without the need for scrolling interaction.

6.4. Layout Convergence

We have observed layouts that do not converge, which is a known problem for various classes of layout algorithms. Influencing factors are the step size, and the initialized forces in particular. If the latter are too extreme, the mass-spring model is not able to approach the local optimum position for a data point due to the step size of the simulation. We approached this problem by introducing a damping factor, which monotonically decreases the simulation step size over time leading to less movement and better converging towards optimal positions. A future work approach would be to learn a machine learning model that predicts layout convergence, which in turn, can be used as a guidance component for parameter setting and tuning. Additionally, a future approach would benefit from instigating how sensitive the layout is for changes in the parameter space, this would provide the user with additional guiding towards stable layouts.

7. Conclusions

We have presented LayoutExOmizer, a visual analytics approach that enables users to steer, analyze, and compare multiple layouts for 2D data points to identify data layouts most useful for a given dataset, task, and user preference. LayoutExOmizer offers a high degree of human control for parameter steering and quality assessment, as well as high levels of computer automation, leading to reliable and trustworthy layouts. With LayoutExOmizer, users can, e.g., face the trade-off between the preservation of positions of data points and the reduction of overplotting interactively and iteratively. In two usage scenarios, we validated the applicability of our approach for different datasets. Future work includes more empirical experiments with user involvement a) to further validate the applicability of the visual interface for different application areas and b) to acquire preference data submitted with user feedback. With the latter, we want to continue the path towards finding a layout method that automatically finds optimal trade-offs for complex data, task, and human-centered optimization goals.

References

[AASB19] ABBAS, MOSTAFA M., AUPETIT, MICHAËL, SEDLMAIR, MICHAEL, and BENSMAIL, HALIMA. "ClustMe: A Visual Quality Measure for Ranking Monochrome Scatterplots based on Cluster Patterns". *Computer Graphics Forum (CGF)* 38.3 (2019), 225–236. DOI: https://doi.org/10.1111/cgf.13684 2.

[Ans73] ANSCOMBE, F. J. "Graphs in Statistical Analysis". *The American Statistician* 27.1 (1973), 17–21. DOI: 10.1080/00031305.1973.10478966 1.

[APSN13] ACHARYA, ANITA S, PRAKASH, ANUPAM, SAXENA, PIKEE, and NIGAM, ARUNA. "Sampling: Why and how of it". *Indian Journal of Medical Specialties* 4.2 (2013), 330–333 3.

[BKC*13] BORGO, RITA, KEHRER, JOHANNES, CHUNG, DAVID H. S., et al. "Glyph-based Visualization: Foundations, Design Guidelines, Techniques and Applications". *Eurographics - State of the Art Reports*. Eurographics, 2013, 39–63. DOI: 10.2312/conf/EG2013/stars/039-063 2.

[BM10] BRUCKNER, STEFAN and MÖLLER, TORSTEN. "Result-Driven Exploration of Simulation Parameter Spaces for Visual Effects Design". *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 16.6 (2010), 1468–1476. DOI: 10.1109/TVCG.2010.190 3.

[BMPM12] BOOSHEHRIAN, MARYAM, MÖLLER, TORSTEN, PETERMAN, RANDALL M., and MUNZNER, TAMARA. "Vismon: Facilitating Analysis of Trade-Offs, Uncertainty, and Sensitivity In Fisheries Management Decision Making". *Computer Graphics Forum (CGF)* 31.3 (2012), 1235–1244. DOI: 10.1111/j.1467-8659.2012.03116.x 3.

[BTK11] BERTINI, ENRICO, TATU, ANDRADA, and KEIM, DANIEL. "Quality metrics in high-dimensional data visualization: An overview and systematization". *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 17.12 (2011), 2203–2212 2, 3.

[BvLBS11] BERNARD, JÜRGEN, von LANDESBERGER, TATIANA, BREMM, SEBASTIAN, and SCHRECK, TOBIAS. "Multiscale visual quality assessment for cluster analysis with Self-Organizing Maps". *Visualization and Data Analysis (VDA)*. SPIE Press, 2011, 78680N.1–78680N.12. DOI: 10.1117/12.872545 2.

[CPPS20] CHEN, FATI, PICCININI, LAURENT, PONCELET, PASCAL, and SALLABERRY, ARNAUD. "Node Overlap Removal Algorithms: an Extended Comparative Study". *Journal of Graph Algorithms and Applications* 24.4 (2020), 683–706. DOI: 10.7155/jgaa.00532 1, 2.

[EMK*21] ESPADOTO, MATEUS, MARTINS, RAFAEL MESSIAS, KERREN, ANDREAS, et al. "Toward a Quantitative Survey of Dimension Reduction Techniques". *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 27.3 (2021), 2153–2173. DOI: 10.1109/TVCG.2019.2944182 1, 2.

[FR91] FRUCHTERMAN, THOMAS M. J. and REINGOLD, EDWARD M. "Graph drawing by force-directed placement". *Software: Practice and Experience* 21.11 (1991), 1129–1164. DOI: https://doi.org/10.1002/spe.4380211102 3.

[GKN05] GANSNER, EMDEN R., KOREN, YEHUDA, and NORTH, STEPHEN. "Graph Drawing by Stress Majorization". *Graph Drawing*. Springer, 2005, 239–250. ISBN: 978-3-540-31843-9 3.

[GM94] GARLAND, KEN and MAP, MR BECK'S UNDERGROUND. "Capital Transport Publishing". *Middlesex, UK* (1994) 2.

[HIMF98] HAYASHI, KUNIHIKO, INOUE, MICHIKO, MASUZAWA, TOSHIMITSU, and FUJIWARA, HIDEO. "A Layout Adjustment Problem for Disjoint Rectangles Preserving Orthogonal Order". *Graph Drawing*. Springer, 1998, 183–197. ISBN: 978-3-540-37623-1 3.

[JRHT14] JIANU, RADU, RUSU, ADRIAN, HU, YIFAN, and TAGGART, DOUGLAS. "How to Display Group Information on Node-Link Diagrams: An Evaluation". *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 20.11 (2014), 1530–1541. DOI: 10.1109/TVCG.2014.2315995 2.

[JZF*09] JEONG, DONG HYUN, ZIEMKIEWICZ, CAROLINE, FISHER, BRIAN D., et al. "iPCA: An Interactive System for PCA-based Visual Analytics". *Computer Graphics Forum (CGF)* 28.3 (2009), 767–774. DOI: 10.1111/j.1467-8659.2009.01475.x 3.

[Kru64] KRUSKAL, JOSEPH B. "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis". *Psychometrika* 29.1 (1964), 1–27 2, 7.

[LV09] LEE, JOHN A. and VERLEYSEN, MICHEL. "Quality assessment of dimensionality reduction: Rank-based criteria". *Neurocomputing* 72.7 (2009). Advances in Machine Learning and Computational Intelligence, 1431–1443. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2008.12.017 3.

[LV10] LEE, JOHN ALDO and VERLEYSEN, MICHEL. "Scale-independent quality criteria for dimensionality reduction". *Pattern Recognit. Lett.* 31.14 (2010), 2248–2257. DOI: 10.1016/j.patrec.2010.04.013 3.

[Lyo96] LYONS, KELLY A. "Cluster busting in anchored graph drawing". PhD thesis. Ph. D. thesis, Department of Computing & Information Science Queen's University, 1996 3.

[Mac04] MACEACHREN, ALAN M. *How Maps Work - Representation, Visualization, and Design*. Guilford Press, 2004. ISBN: 978-1-57230-040-8. URL: http://www.guilford.com/cgi-bin/cartscript.cgi?page=pr/maceachren.htm 1.

[MF17] MATEJKA, JUSTIN and FITZMAURICE, GEORGE W. "Same Stats, Different Graphs: Generating Datasets with Varied Appearance and Identical Statistics through Simulated Annealing". *Conference on Human Factors in Computing Systems (CHI)*. ACM, 2017, 1290–1294. DOI: 10.1145/3025453.3025912 1.

[Mon10] MONMONIER, MARK. *Rhumb lines and map wars: A social history of the Mercator projection*. University of Chicago Press, 2010 2.

[MPOW17] MICALLEF, LUANA, PALMAS, GREGORIO, OULASVIRTA, ANTTI, and WEINKAUF, TINO. "Towards Perceptual Optimization of the Visual Design of Scatterplots". *IEEE Transactions on Visualization and Computer Graphics* 23.6 (2017), 1588–1599. DOI: 10.1109/TVCG.2017.2674978 3.

[Mun14] MUNZNER, TAMARA. *Visualization Analysis and Design*. A.K. Peters visualization series. A K Peters, 2014. ISBN: 978-1-466-50891-0. URL: http://www.cs.ubc.ca/%5C%7Etmm/vadbook/ 2.

[MvGBW19] MUMTAZ, HARIS, van GARDEREN, MEREKE, BECK, FABIAN, and WEISKOPF, DANIEL. "Label Placement for Outliers in Scatterplots." *Conference on Visualization (EuroVis) - Short Papers*. 2019, 1–5 2.

[NL20] NGO, QUYNH QUANG and LINSEN, LARS. "Interactive Generation of 1D Embeddings from 2D Multi-dimensional Data Projections". *25th International Symposium on Vision, Modeling and Visualization, VMV 2020, Tübingen, Germany, September 28 - October 1, 2020*. Eurographics Association, 2020, 79–87. DOI: 10.2312/vmv.20201190 3.

[Noa03] NOACK, ANDREAS. "An Energy Model for Visual Graph Clustering". *Graph Drawing*. Vol. 2912. Lecture Notes in Computer Science. Springer, 2003, 425–436. DOI: 10.1007/978-3-540-24595-7_40 3.

[PST*17] PAJER, STEPHAN, STREIT, MARC, TORSNEY-WEIR, THOMAS, et al. "WeightLifter: Visual Weight Space Exploration for Multi-Criteria Decision Making". *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 23.1 (2017), 611–620. DOI: 10.1109/TVCG.2016.2598589 3.

[SA15] SEDLMAIR, M. and AUPETIT, M. "Data-driven Evaluation of Visual Quality Measures". *Computer Graphics Forum (CGF)* 34.3 (2015), 201–210. DOI: https://doi.org/10.1111/cgf.12632 3.

[SBMK14] STEIGER, MARTIN, BERNARD, JÜRGEN, MAY, THORSTEN, and KOHLHAMMER, JÖRN. "A Survey of Direction-preserving Layout Strategies". *Spring Conference on Computer Graphics*. SCCG. Smolenice, Slovakia: ACM, 2014, 21–28. ISBN: 978-1-4503-3070-1. DOI: 10.1145/2643188.2643189 2.

[SHB*14] SEDLMAIR, MICHAEL, HEINZL, CHRISTOPH, BRUCKNER, STEFAN, et al. "Visual Parameter Space Analysis: A Conceptual Framework". *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 20.12 (2014), 2161–2170. DOI: 10.1109/TVCG.2014.2346321 3.

[Shn20] SHNEIDERMAN, BEN. "Human-Centered Artificial Intelligence: Reliable, Safe & Trustworthy". *Int. J. Hum. Comput. Interact.* 36.6 (2020), 495–504. DOI: 10.1080/10447318.2020.1741118 2.

[SSS*14] SACHA, DOMINIK, STOFFEL, ANDREAS, STOFFEL, FLORIAN, et al. "Knowledge Generation Model for Visual Analytics". *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 20.12 (2014), 1604–1613. DOI: 10.1109/TVCG.2014.2346481 3.

[SZS*17] SACHA, DOMINIK, ZHANG, LEISHI, SEDLMAIR, MICHAEL, et al. "Visual Interaction with Dimensionality Reduction: A Structured Literature Analysis". *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 23.1 (2017), 241–250. DOI: 10.1109/TVCG.2016.2598495 3.

[TG83] TUFTE, EDWARD R and GRAVES-MORRIS, PETER R. *The visual display of quantitative information*. Vol. 2. 9. Graphics press Cheshire, CT, 1983 3.

[TSM*11] TORSNEY-WEIR, THOMAS, SAAD, AHMED, MÖLLER, TORSTEN, et al. "Tuner: Principled Parameter Finding for Image Segmentation Algorithms Using Visual Response Surface Exploration". *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 17.12 (2011), 1892–1901. DOI: 10.1109/TVCG.2011.248 3.

[VBW17] VEHLOW, CORINNA, BECK, FABIAN, and WEISKOPF, DANIEL. "Visualizing group structures in graphs: A survey". 36.6 (2017), 201–225 1, 2.

[vdMH08] Van der MAATEN, LAURENS and HINTON, GEOFFREY. "Visualizing Data using t-SNE". *Journal of Machine Learning Research* 9.86 (2008), 2579–2605. URL: http://jmlr.org/papers/v9/vandermaaten08a.html 2, 3.

[vGPNB17] Van GARDEREN, M., PAMPEL, B., NOCAJ, A., and BRANDES, U. "Minimum-Displacement Overlap Removal for Geo-referenced Data Visualization". *Computer Graphics Forum (CGF)* 36.3 (2017), 423–433. DOI: https://doi.org/10.1111/cgf.13199 2.

[VK06] VENNA, JARKKO and KASKI, SAMUEL. "Local multidimensional scaling". *Neural Networks* 19.6-7 (2006), 889–899. DOI: 10.1016/j.neunet.2006.05.014 2.

[VPN*10] VENNA, JARKKO, PELTONEN, JAAKKO, NYBO, KRISTIAN, et al. "Information Retrieval Perspective to Nonlinear Dimensionality Reduction for Data Visualization". *J. Mach. Learn. Res.* 11 (2010), 451–490. URL: https://dl.acm.org/citation.cfm?id=1756019 2.

[WAG06] WILKINSON, L., ANAND, A., and GROSSMAN, R. "High-Dimensional Visual Analytics: Interactive Exploration Guided by Pairwise Views of Point Distributions". *IEEE Transactions on Visualization and Computer Graphics (TVCG)* 12.6 (2006), 1363–1372. DOI: 10.1109/TVCG.2006.94 2.

[WZM*19] WÖHLER, LESLIE, ZOU, YUXIN, MÜHLHAUSEN, MORITZ, et al. "Learning a Perceptual Quality Metric for Correlation in Scatterplots". *24th International Symposium on Vision, Modeling, and Visualization, VMV 2019, Rostock, Germany, September 30 - October 2, 2019*. Eurographics Association, 2019, 55–62. DOI: 10.2312/vmv.20191318 2.