Evaluating Reordering Strategies for Cluster Identification in Parallel Coordinates

Michael Blumenschein[1] , Xuan Zhang[2] , David Pomerenke[1] , Daniel A. Keim[1] , and Johannes Fuchs[1]

[1]University of Konstanz, Germany   [2]RWTH Aachen University, Germany

(a) *Medium clutter, Similar*   **(b)** *High clutter, Similar*   **(c)** *Medium clutter, Dissimilar*   **(d)** *High clutter, Dissimilar*
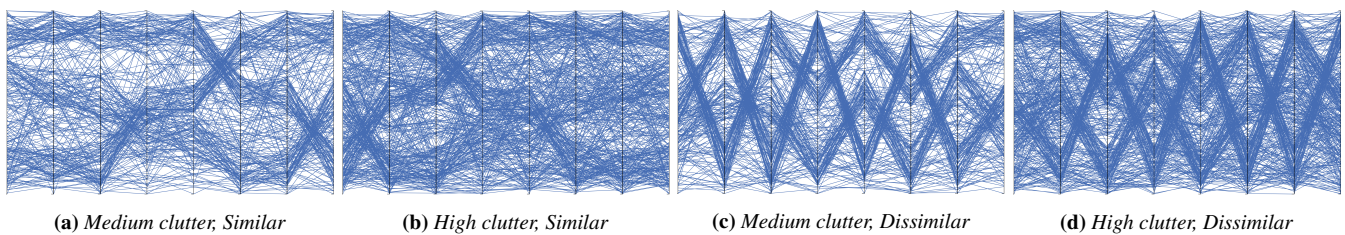
Figure 1: *A dataset with three clusters and two different clutter levels is sorted by* similarity *(a–b) and* dissimilarity *(c–d) of neighboring axes pairs. Clusters are more salient when arranging dissimilar dimensions next to each other. We show that in cluttered datasets, participants are more accurate and more confident when performing cluster identification tasks on such a layout.*

Abstract
The ability to perceive patterns in parallel coordinates plots (PCPs) is heavily influenced by the ordering of the dimensions. While the community has proposed over 30 automatic ordering strategies, we still lack empirical guidance for choosing an appropriate strategy for a given task. In this paper, we first propose a classification of tasks and patterns and analyze which PCP reordering strategies help in detecting them. Based on our classification, we then conduct an empirical user study with 31 participants to evaluate reordering strategies for cluster identification tasks. We particularly measure time, identification quality, and the users' confidence for two different strategies using both synthetic and real-world datasets. Our results show that, somewhat unexpectedly, participants tend to focus on dissimilar rather than similar dimension pairs when detecting clusters, and are more confident in their answers. This is especially true when increasing the amount of clutter in the data. As a result of these findings, we propose a new reordering strategy based on the dissimilarity of neighboring dimension pairs.

CCS Concepts
• *Human-centered computing* → *Empirical studies in visualization;*

1. Introduction

Parallel coordinates plots (PCPs) [Ins85, Ins09b] are a popular and well-researched technique to visualize multi-dimensional data. Dimensions are represented by vertical, equally spaced axes. Data records are encoded by polylines, connecting the respective values on each axis. PCPs have been applied to practical applications of various domains [JF16], such as finance [AZZ10], traffic safety [FWR99], and network analysis [SC*05]. As discussed by Andrienko and Andrienko [AA01], PCPs are suited for a multitude of analysis tasks, such as cluster, correlation, and outlier analysis.

Compared to other visualizations for multi-dimensional data (e.g., RadVis [HGM*97], MDS and PCA projections, scatter plots, and scatter plot matrices), PCPs have the advantage to trace data records and patterns across a large set of dimensions. Empirical studies have shown that PCPs outperform scatter plots in clustering tasks, outlier, and change detection [KAC15], but are less suited for correlation analysis [LMvW10, HYFC14] and value retrieval [KZZM12].

A major challenge of visualizations is *visual clutter*, which in-

fluences the perception of visible patterns [SKKAJ15]. This problem is particularly given in PCPs, as line crossing and overplotting distort salient structures. Therefore, the community has proposed a multitude of enhancements, such as sampling [ED06], edge bundling [MM08], interactive highlighting [MW95], and the usage of transparency [JLJC05] to reduce the impact of visual clutter.

The ordering of axes plays a fundamental role in the design of a PCP and has a strong effect on the overall pattern structure [JJ09]. In contrast to data preprocessing, sampling, dimension filtering, and other enhancements, reordering does not remove data from the PCP [PWR04, PL17], but changes the visual structure among neighboring axes. Depending on the user's analysis goal, some patterns are more interesting than others [DK10]. As a result, more than 30 different ordering strategies have been developed by the community to support a multitude of tasks. Some of these strategies group similar dimension pairs [ABK98], try to avoid line crossings [DK10], or put the most important dimensions first [YPWR03]. However, our community lacks empirical guidance and recommendations for

choosing an appropriate strategy for a given task [BBK*18]. In this paper, we address this limitation by summarizing the state-of-the-art in axes reordering strategies, as well as presenting a first empirical user study that measures the performance of two ordering methods for cluster identification in PCPs. Our study focuses on cluster analysis as the majority of ordering strategies are designed for this task.

We claim two main contributions. First, we provide guidance in selecting reordering strategies based on their intended patterns. Many existing algorithms follow similar concepts but differ in their implementation. To support users, we introduce a classification of the existing layout algorithms, group them according to their inner workings, and summarize their intended patterns and meta-characteristics. For more practical support, we implemented a set of 14 strategies in JavaScript and made them along with the source code available on our website for testing: http://subspace.dbvis.de/pcp.

Second, we measure the performance of two reordering strategies for cluster identification tasks by an empirical user study with 31 participants. We realized that the often proposed similarity-based axes arrangement (e.g., [ABK98, YPWR03, AdOL06]) is not always the most effective solution to identify clusters. As shown in Figure 1, arranging axes with a high dissimilarity next to each other produces more salient clusters, in particular in cluttered and noisy datasets. A reason for this effect is that lines with strong slopes are moving closer together, making clusters visually more prominent [PDK*19]. To find out whether this arrangement is more useful than a similarity-based layout, we conducted a user study and measured performance with respect to cluster quality, completion time, and users' confidence using synthetic and real-world datasets. Our results show that participants tend to focus on dissimilar axes pairs when selecting clusters and are more accurate and confident when doing so.

As a secondary contribution, we provide a benchmark dataset with 82 synthetic and real-world datasets for clustering analysis. For reproducibility, we make all our material and results, statistical analysis, and source code publicly available at https://osf.io/zwm69.

The remainder of this paper is structured as follows: In the next section, we summarize important related work. Then, in Section 3, we survey existing reordering strategies for PCPs and classify them based on their intended patterns and inner workings (first contribution). Afterwards, in Section 4, we describe our user study design and report the statistical analysis results in Section 5 (second contribution). Finally, we discuss our findings and derive design considerations for axes orderings in cluster identification tasks.

2. Background and Related Work

In the following, we summarize the challenges of axes reordering, the results of existing user studies, and the relation of automatic ordering to interactive and semi-automatic analysis support.

2.1. Challenges of Axes Reordering of Parallel Coordinates

Linear ordering of an $n-$dimensional dataset in PCPs faces two main challenges. First, computing and evaluating all dimension permutations is computationally expensive. Ankerst et al. [ABK98] show that the ordering of axes according to some useful objective function is NP-complete. Therefore, the exhaustive search for a useful ordering is tedious, even for a modest number of dimensions [PWR04]. Second, the usefulness of a particular ordering highly depends on the analysis task of the user [DK10, PL17], and is influenced by the

complexity of the data [TAE*11]. More importantly, optimizing the axes ordering to highlight a particular pattern may even obstruct other patterns [JJ09], which are of relevance in a different scenario. Therefore, it is vital to carefully choose an appropriate strategy to arrange the axes in parallel coordinates.

More than 30 reordering strategies have been developed (see Section 3), many of which follow similar concepts but differ in their implementation affecting, for example, the runtime and quality of the results. Quality metrics and layout algorithms for PCPs have been summarized before: Heinrich and Weiskopf [HW13] give a comprehensive overview of the state-of-the-art for PCP research, including manual and automatic reordering approaches. Bertini et al. [BTK11] and Behrisch et al. [BBK*18] summarize quality metrics to optimize the visual representation. Ellis and Dix [ED07] discuss reordering from a clutter perspective. While Behrisch et al. [BBK*18] group the quality metrics by their analysis task, the literature still misses a summary of the different PCP patterns and a discussion on which reordering algorithms favor or avoid particular patterns [TAE*11]. In our paper, we close this gap by introducing a classification along with a characterization of the reordering algorithms.

2.2. Evaluation of Axes Reorderings and Empirical Studies

There is a lack of empirical studies to measure the performance of specific axes orderings for different analysis tasks [JF16]. Most strategies are "evaluated" using examples of synthetic or real-world data (see Table 1) instead of comparing it to previous approaches. Exceptions are the works by Ferdosi & Roerdink [FR11] and Tatu et al. [TAE*11], which compare the resulting orders with competing approaches. However, no feedback from real users is provided.

Many reorderings claim to be useful for cluster analysis, but we do not know yet which patterns are most effective in identifying clusters. There is no user study that compares different reorderings for cluster identification in particular or different analysis tasks in general. Therefore, we want to push PCP reordering towards an empirically-driven research field by evaluating two axes reordering techniques for cluster identification. The works most closely related to ours are the empirical studies by Holten & van Wijk [HvW10] (measuring response time and cluster identification correctness for nine PCP variations), Kanjanabose et al. [KAC15] (measuring response time and clustering accuracy in PCP, scatter plots, and classical tables), and the study by Johansson et al. [JFLC08] evaluating clutter threshold for the identification of patterns. However, none of these studies consider different axes orderings as an independent variable.

2.3. Relation to Interactive and Semi-Automatic Analysis

Besides axes reordering, countless enhancements have been developed to support the understanding of patterns in parallel coordinates. A comprehensive overview is out of the scope of this paper but can be found in the surveys by Ellis & Dix [ED07], Bertini et al. [BTK11], Heinrich & Weiskopf [HW13], and Behrisch et al. [BBK*18]. Many techniques involve users within an interactive exploration workflow or combine the representation with automatic algorithms for pattern detection. Examples are the usage of clustering algorithms [FWR99, JLJC05, Mou11], automatic sampling techniques [ED06], and interactive highlighting [MW95]. Inselberg [Ins09a], and Hurley & Oldford [HO10] propose to clone
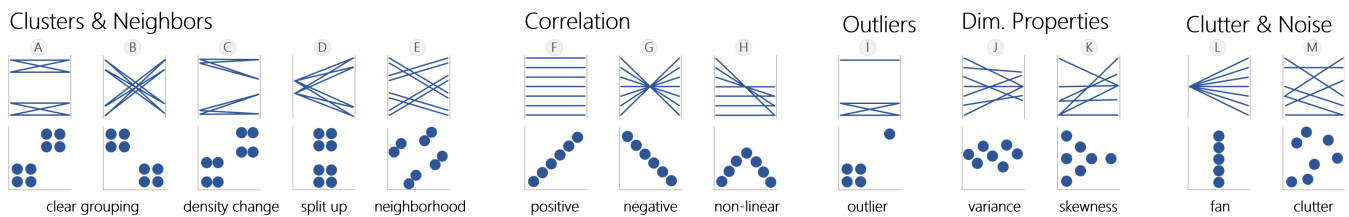
Figure 2: *Comparison of visual patterns in parallel coordinates and their scatter plot representation.*

and arrange dimensions such that all pairwise permutations are visible, similar to a scatter plot matrix. Based on this initial view, the user can then start the exploration.

While the usefulness of such interactive and visual analytics approaches have been shown in many user studies, they are facing two challenges: First, most algorithms rely on sensitive parameters which influence the quality of the result. For example, k-means clustering [HKP11] requires the number of clusters as user input, which is typically unknown for a new dataset. Second, interactive exploration and highlighting are difficult if users do not know what they are searching for, and the initial configuration of a PCP does not show (parts of) interesting patterns. Often, this results in trial-and-error interactions, in which patterns are only detected 'by accident'. This is particularly true if the dataset contains a large number of dimensions, and relevant patterns only exist in smaller subspaces.

Therefore, we need methods to give analysts good starting conditions for their (interactive) analysis. Hereby, an important aspect is the arrangement of axes, as it significantly changes the visual patterns among neighboring axes [JJ09]. In our paper, we provide a categorization of reordering algorithms and their intended patterns, which helps analysts to make an educated selection.

3. Classification of Reordering Strategies

This section addresses two questions: *Which patterns are emphasized by which reordering strategy?* And *which algorithms have been implemented to solve the reordering problem?* Before answering these questions, we provide an overview of important patterns.

3.1. Visual Patterns

Figure 2 shows five groups of the most common patterns in parallel coordinates and their representation in scatter plots:

Clusters & Neighbors Ⓐ – Ⓔ. Typical cluster structures show one or more groups of dense lines in a similar direction. While Ⓐ & Ⓑ seem similar in scatter plots, the visible structure in PCP differs significantly. Ⓒ shows clusters that change their density (cluster compactness) and Ⓓ, a cluster that splits up into sub-clusters.

Structures, preserving neighborhood information, are a special case of clusters. A (small) set of data records similar (close) to each other in one dimension are also similar in the neighboring dimension, which results in groups of parallel lines Ⓔ.

Correlations Ⓕ – Ⓗ. Positive and negative correlations look similar in a scatter plot. However, the PCP patterns differ: lines are parallel for positive Ⓕ, and in a star-like pattern for negative correlations Ⓖ. Variations of non-linear correlations may look different in both scatter plots and PCPs. Ⓗ shows only an example as the pattern depends on the type and degree of change in both dimensions.

Outliers Ⓘ. Outliers squeeze the majority of PCP lines together, resulting in a cluster-like pattern, hiding the underlying structure.

Dimension Properties Ⓙ – Ⓚ. show patterns of dimensions, ordered by variance and skewness. The lines' slope indicates whether patterns stay consistent (parallel) or change across axes.

Clutter & Noise. Randomly distributed data without a clear pattern is considered noise or clutter *in the data*. The lines in the PCP cross without any particular structure Ⓜ. The fan pattern Ⓛ describes a cluster transitioning into clutter, a special case of density change Ⓒ.

Our selection of patterns is based on the work by Dasgupta & Kosara [DK10], Wegman [Weg90], Heinrich & Weiskopf [HW13, HW15], and Zhou et al. [ZYQ*08]. The cluster variations (i.e., patterns Ⓐ – Ⓓ) are inspired by Pattern Trails [JHB*17], which introduces a taxonomy of pattern transitions between multi-dimensional subspaces. These patterns can be adapted to PCP, as two neighboring axes show a transition between one-dimensional subspaces. Finally, we add variance Ⓙ and skewness Ⓚ, which is produced by the algorithms described in [LHZ16, SCC*18, YPWR03]. We limit our patterns to 2D PCPs and ignore patterns in 3D PCPs (e.g., discussed in [PL17, AKSZ13]. We also consider only patterns among the two axes. Multi-dimensional patterns can be achieved by concatenating multiple two-dimensional patterns.

3.2. Ordering Strategies

To find ordering strategies, we took the 502 references of recent state-of-the-art reports [HW13, JF16, BBK*18] and combined it with 497 papers resulting from a search on the ACM, IEEE Xplore, EG, and DBLP digital library (see keywords and details in the supplementary material). We recursively scanned references and excluded papers that (1) did not propose an automatic axes ordering strategy (e.g., purely interactive approaches), (2) "just" apply a reordering method which has been proposed before, or (3) approaches which do not focus on "standard" parallel coordinates (e.g., 3D PCPs). Using this approach, we collected 18 papers with 32 different strategies.

Table 1 summarizes all reordering strategies, grouped by their ordering concept: strategies transforming the reordering into an *optimization problem* (Section 3.3), *implementing* efficient or sophisticated *algorithms* (Section 3.4), and approaches focusing on *properties* of *single dimensions* (Section 3.5). For each approach, we indicate the favored patterns, the involved axes, and the performed evaluation to show the usefulness (see caption of Table 1 for details).

3.3. Optimization Problem and Objective Functions

The largest group of reordering strategies transforms the axes arrangement into an optimization problem. These approaches measure the quality of a particular ordering by an objective function, which is then either minimized or maximized by an optimization algorithm.

Table 1: *Reordering Classification summarizing the inner workings of reordering strategies for parallel coordinates. For each technique, we mark if it favors or avoids a particular pattern, if present in the data. Empty cells mean that the technique is not designed for this pattern and produces/avoids it primarily by change. Approaches are grouped by their concept and sorted according to the similarity of patterns that they favor and avoid. Additionally, we indicate the number of dimensions that are taken into account for the reordering and mark techniques which can be combined with dimension filters and subspace analysis. Finally, we mark the evaluation type used in the paper. Due to space limitations, we only show the name of the first author.*

Patterns: algorithm favors ● or avoids ⊗ pattern, or it depends on data properties and algorithm parameters ◐.

Axes considered for ordering: each dimension separately (|), two neighboring dimensions (||), or the majority of dimensions (||||).

The technique can be combined with: dimension filters (DR) and subspace analysis (S).

Evaluation: case study or example Ⓔ, comparison with other techniques ⬄, and empirical study 👤.

	Reordering Technique	Ⓐ	Ⓑ	Ⓒ	Ⓓ	Ⓔ	Ⓕ	Ⓖ	Ⓗ	Ⓘ	Ⓙ	Ⓚ	Ⓛ	Ⓜ	Axes	Dim.	Eval.
Objective Function & Optimization Algorithm	Tatu (no label) [TAE*11]	●	●		●									⊗	||		⬄
	Tatu (given label) [TAE*11]	●	●		◐									⊗	||		⬄
	Long (class consist.) [Van15]	●	●		⊗									⊗	||		Ⓔ
	Zhou (cluster trace) [ZYYC18]	●	●	◐	⊗									⊗	||		Ⓔ
	Peltonen (neighbor) [PL17]	●	●	◐	◐	●							⊗	⊗	||		Ⓔ
	Dasgupta (overpl.) [DK10]	●	●	◐	◐		⊗								||		Ⓔ
	Dasgupta (parall.) [DK10]		⊗				●	⊗						⊗	||		Ⓔ
	Ankerst (similar) [ABK98]	●	⊗				●	⊗						⊗	||		Ⓔ
	Yang (similarity) [YPWR03]	●	⊗				●	⊗						⊗	||	DR +S	Ⓔ
	Xiang (clus. interac.) [XFJ*12]	●	⊗											⊗	||		Ⓔ
	Ankerst (correl.) [ABK98]	●	⊗				●	⊗						⊗	||		Ⓔ
	Artero (clutter) [AdOL06]	◐	⊗				●	⊗					⊗	⊗	||||		Ⓔ
	★ **Blumenschein (dissimil.)**	⊗	●				⊗	●						⊗	||		👤
	Dasgupta (cross.) [DK10]	⊗	◐				⊗	●						◐	||		Ⓔ
	Dasgupta (angle) [DK10]	⊗	◐				⊗	●						◐	||		Ⓔ
	Dasgupta (mut. i.) [DK10]						●	●	●					⊗	||		Ⓔ
	Makwana (struc.) [MTJ12]							●					●	●	||		Ⓔ
	Peng (outlier) [PWR04]									●					||		Ⓔ
	Dasgupta (entropy) [DK10]													⊗	||		Ⓔ
	Dasgupta (diverg.) [DK10]												●	⊗	||		Ⓔ
Reordering Algorithm	Johansson (cluster) [JJ09]	●	●	●	●									⊗	||||	DR	Ⓔ
	Ferdosi (subspace) [FR11]	●	●	●	●									⊗	||||	DR +S	⬄
	Huang (set theory) [HHJ11]	●	●	●										⊗	||		Ⓔ
	Artero (similar) [AdOL06]	●	⊗				●	⊗						⊗	||	DR	Ⓔ
	Johansson (correl.) [JJ09]	●	⊗				●	⊗						⊗	||	DR	Ⓔ
	Lu (non-lin. corr.) [LHH12]						◐	◐	●					⊗	||		Ⓔ
	Johansson (outlier) [JJ09]									●				⊗	||||	DR	Ⓔ
Dimension QM	Schloerke (class) [SCC*18]	●	●	●										⊗	|		−
	Schloerke (outlier) [SCC*18]									●				⊗	|		−
	Lu (svd) [LHZ16]										●	●	●	⊗	||||	DR	Ⓔ
	Yang (variance) [YPWR03]										●	●	●	⊗	|	DR +S	Ⓔ
	Schloerke (skewn.) [SCC*18]											●		⊗	|		−

3.3.1. Objective Functions Measuring Cluster Structures

Tatu et al. [TAE*09,TAE*11] argue that clusters consist of lines with a similar position and direction (patterns Ⓐ, Ⓑ, and Ⓓ). The authors take a rendered image of a PCP and apply a Hough space transformation [Hou62]. Each PCP line segment is mapped into one point within the Hough space. The point's location represents the position and slope of the line segment. The objective function measures dense areas (clusters) of points in the Hough space. Long [Van15] first computes a centroid for all given clusters. Then for each data record, the nearest centroid is identified (using the area between the lines as similarity function), and the objective function measures the ratio of correctly classified records. Cluster patterns Ⓐ and Ⓑ are highlighted, while a cluster split Ⓓ is avoided. Zhou et al. [ZYYC18] aim at clusters that can be followed across neighboring axes (Ⓐ and Ⓑ). They compute a hierarchical clustering on every dimension and use the cluster similarity as quality. Dasgupta & Kosara [DK10] introduce seven different metrics, known as *pargnostics*. A metric aiming for clusters like Ⓐ and Ⓑ is *overplotting*. It measures the number of pixels that are not visible due to overlapping lines. When maximizing this measure, there is a high information loss, but a high-density of lines, i.e., clusters. Finally, Xiang et al. [XFJ*12] try to avoid intersecting clusters Ⓑ by measuring the crossing of clusters among axes. This results in horizontal cluster structures Ⓐ.

Peltonen & Lin [PL17] aim to preserve the neighborhood distribution of records (pattern Ⓔ). The objective function measures the similarity of nearest neighbors for all data records across two dimensions. Clusters like Ⓐ and Ⓑ are a special case of neighborhood relationships and are therefore considered as well.

3.3.2. Similarity-based Metrics for Clusters and Correlation

The main idea of the following approaches is to arrange similar dimensions next to each other. This results in cluster- and correlation patterns. The definition of similarity differs across the techniques. Ankerst et al. [ABK98] use a Euclidean distance and Pearson correlation for the similarity computation. The approach by Yang et al. [YPWR03] follows the same idea. However, they structure the dimensions into a hierarchy using a hierarchical clustering algorithm to highlight also clusters in subspaces of the dataset. The hierarchical structure also helps to speed up the computation time, as each subtree can be sorted independently. Depending on the similarity function, and whether the objective function is minimized or maximized, these approaches aim for the patterns Ⓐ, Ⓑ, Ⓕ, and Ⓖ.

Other metrics try to order axes such that lines are most parallel or diverging a lot. These patterns can help to identify correlations, but may also favor clusters to some extent. Artero et al. [AdOL06] propose the total length of poly-lines as metric for pattern Ⓕ. Similarly, there are four pargnostic [DK10] measures: (1) *Maximize the number of line crossings* to identify inverse relationships Ⓖ in the data. (2) *Maximizing Ⓖ the angle of crossings*. (3) *Maximizing parallelism*, resulting in less cluttered PCPs, which highlight positive correlations Ⓕ. (4) *Maximizing the mutual information*, which measures the dependency between variables, i.e., optimizing for positive Ⓕ, negative Ⓖ, and non-linear correlations Ⓗ.

3.3.3. Objective Functions for Clutter and Outliers

The pargnostic metric *maximizing divergence* results in fan pattern Ⓛ, which helps to identify cluster-to-noise relationships. *Maximizing the entropy of neighboring axes* corresponds to a high informa-tion density, highlighting many line crossings and inverse relationships. The metric does not favor specific patterns, but results in busy, but very readable charts, according to the authors [DK10]. The metric by Makwana et al. [MTJ12] differs from previous metrics. The authors propose to order dimensions such that neighboring axes contain lines with different slopes, resulting in cluttered Ⓜ PCPs.

Peng et al. [PWR04] interpret outliers as data points that do not belong to a cluster. They measure the ratio of outliers against the number of data points. When maximized, outliers are highlighted (pattern Ⓘ), when minimized, outliers will not be highlighted.

3.3.4. Optimization Algorithms for Objective Functions

Except for [AdOL06], all approaches measure the quality between neighboring axes ‖ and use the average as the objective function. To minimize or maximize this function, various heuristics are applied: Random swapping (particularly useful for very large datasets) [YPWR03, PWR04], Ant-optimization [ABK98], A*Search [TAE*09, TAE*11], Nearest-neighbor-based [PWR04], Branch and bound optimization [DK10, MTJ12], Non-linear optimization algorithm [PL17], and Backtracking [ZYYC18].

3.4. Reordering by Algorithms

The second class of strategies arranges dimensions based on layout algorithms. Compared to optimization procedures, this has two advantages: (1) Algorithms which approximate the understanding of an objective function, lead to more efficient but potentially less accurate results (e.g., [LHH12, AdOL06, JJ09]). (2) Objective functions are typically defined only between neighboring axes. Using more advanced algorithms (e.g., based on subspace clustering) lead to PCP, which aims for higher-dimensional patterns [FR11].

3.4.1. Algorithms for Efficient Reordering

Artero et al. [AdOL06] and Johansson & Johansson [JJ09] speed up the similarity and correlation-based axes arrangement, originally presented by Ankerst et al. [ABK98]. Both algorithms are identical, except for the similarity function. Artero et al. use a Euclidean distance, Johansson & Johansson, a Pearson correlation coefficient. The algorithm starts with the most similar dimension pair in the center of the PCP. Iteratively, the next most similar dimension is appended to the left or right side. While this approach is efficient, it also has the advantage that the most salient structure (the most similar dimensions) typically ends up close to the center of the PCP, which users are most attracted to [NVE*17]. Lu et al.'s approach [LHH12] orders dimensions based on correlation. They use the nonlinear correlation coefficient (NCC), which is sensitive to any relationship Ⓕ–Ⓗ (not only linear ones) and can be used for partial similarity matching as well [ABK98]. The proposed algorithm combines the ordering by (non-linear) correlations together with an importance driven arrangement. The first (left) axis in the PCP is chosen based on the highest singular value after a singular value decomposition (SVD, highest contribution of the dataset). Afterwards, all dimensions are arranged from left to right according to their similarity of the NCC. The approach by Huang et al. [HHJ11] maximizes the uniform line crossings of clusters. Their approach is based on Rough Set Theory [Paw12], and the algorithm sorts the dimensions based on alternating sizes of high and low cardinality of the equivalence classes, leading to cluster patterns Ⓐ–Ⓒ.

3.4.2. Subspace Algorithms for Higher-dimensional Structures

Ferdosi & Roerdink [FR11] use a subspace search algorithm [FBT*10] to identify higher-dimensional clusters with patterns Ⓐ–Ⓓ. The quality of one subspace is based on a density distribution. Subspaces containing multiple clusters that are clearly separated are considered of high quality. First, the algorithm computes all one-dimensional subspaces and arranges the one with the highest quality on the very left of the PCP. Afterwards, all two-dimensional subspaces, which contain the first subspace, are computed, and the highest rank is attached as the second axis. The algorithm continues until all dimensions are part of the PCP, or no more subspace can be computed. Johansson & Johansson [JJ09] apply the MAFIA algorithm [NGC01], resulting in a set of subspaces, along with cluster structures and quality measures. The ordering algorithm then finds the longest sequence of connected variables shared by all detected subspace clusters. It starts with all dimensions of the first subspace (no specific ordering). Further subspaces are iteratively added based on their quality, but only if they share a substantial set of dimensions with the current PCP. The authors use the same algorithm to identify patterns with (multi-dimensional) outliers (pattern Ⓘ).

3.5. Reordering by Dimension-wise Quality Metrics

The third group of reordering techniques computes a quality for each dimension separately (Ⅲ) and sort the axes accordingly. Assuming the quality can be computed efficiently, reordering can be done in linear time. The techniques can also be extended by dimension filtering (e.g., considering only dimensions with a quality above a threshold). Relations *between dimensions* are not considered. Therefore, patterns may be scattered in different parts of the PCP [PL17].

Lu et al. [LHZ16] sort the axes based on each of their contributions to the dataset. They compute an SVD and sort the dimensions according to their singular values. Yang et al. [YPWR03] propose a similar approach but sort the dimensions by variance. Both reorderings result in similar patterns (Ⓙ – Ⓛ). However, Lu et al.'s approach takes the distribution of the entire dataset into account.

Schloerke et al. [SCC*18] propose three different dimension metrics: (1) They use skewness for reordering, resulting in a Ⓚ pattern. (2) The dimensions are sorted by one of the Scagnostics [WAG05] measures. In particular, the *Outlying* measure is useful to highlight outliers in the data (pattern Ⓘ). (3) Finally, the authors order the dimensions such that existing clusters or classes are separated as well as possible. They compute an ANOVA on every dimension based on a given set of class labels and order the dimensions based on the F-statistic. Intuitively, the dimensions are ordered according to how well the given clusters are separated (patterns Ⓐ – Ⓒ).

3.6. Summary

In Table 1, we provide an overview of 32 different reordering approaches to arrange the axes of parallel coordinates. During our analysis, we made a few observations: (1) Many reordering algorithms follow similar concepts, but differ in their implementation and the applied metric. The main reason for this is that axes reordering is computationally complex, and more efficient approaches are necessary for interactive applications. (2) There seems to be a different understanding of the most important area in a PCP. While some reordering approaches try to put the most important dimensions upfront, others try to arrange them in the center. This is in line with

the study by Netzel et al. [NVE*17], who found out that people pay the most attention to the center part of a PCP. (3) The evaluation of novel reordering algorithms is primarily achieved by use cases and example applications. We are not aware of empirical user studies that compare different orderings for a particular analysis task. With our paper, we want to close this gap and provide the first empirical study to evaluate ordering approaches for a particular analysis task.

4. User Study Design: Reordering for Cluster Identification

Our reordering classification in Table 1 reveals that the majority of strategies are designed to support cluster analysis. Therefore, we select this task as the focus of our user study. In particular, we want to assess the performance of *cluster identification*, as this is the foundation for more sophisticated clustering analyses.

For cluster analysis, *similarity-based layouts* are proposed most often. Clusters, if present, can be followed across many axes, as algorithms try to minimize their variance. However, this strategy does not necessarily highlight clusters. This is especially true if the dataset contains noise or clutter, as shown in Figure 1a. While we can identify the clusters, they are visually less salient. In this paper, we define the term *clutter* as data records that do not contribute to a particular pattern (e.g., randomly distributed), often also called *noise*. Cluttered datasets often end up in *visual cluttered* PCP due to many line crossings and overplotting.

Due to experiments with our implemented reordering algorithms, we realized that polylines and clusters with *strong slopes* are visually more prominent than horizontal ones. There are two reasons, as discussed by Pomerenke et al. [PDK*19]: (1) With an increasing slope, the distance between polylines decreases, and less whitespace (background) is visible. Hence, neighboring lines have higher contrast. (2) Compared to horizontal lines, diagonal lines need more pixels to encode a single data point, resulting in a low data-to-ink-ratio [Tuf01]. Both geometric effects make sure that neighboring lines are more easily perceived as a group or cluster. Interestingly, strong slopes are produced when dimensions are ordered by *dissimilarity*. An example can be found in Figure 1c. It shows the same data as in 1a, but with strong slopes due to reordering. Often this results in a zig-zag-like pattern, which makes the visual representation more complex but also ends up in more salient cluster structures.

4.1. Hypotheses

We address the question, *'whether there is a difference between a similarity-based* (SIM) *and dissimilarity-based* (DIS) *axes ordering for a cluster identification task'*. If yes, *'which ordering should be used, and why?'* As the majority of real-world datasets contain noise and clutter, we also want to investigate its influence generally, and in combination with the axes ordering. Hence, we analyze two independent variables: *ordering method* and *clutter level*.

To measure the performance, we use three dependent variables: (i) *time* to identify clusters, (ii) quality of manually *selected clusters* based on similarity to ground truth clusters, and (iii) the *confidence* of the users after the cluster identification. Additionally, we analyze axes-pairs which help to identify the clusters. In particular, we investigate whether users select clusters in similar or dissimilar axes pairs. For our study, we formulate the following three hypotheses:

H1. With an *increasing amount of clutter*, the cluster identification performance drops (independent of the ordering) as cluster structures

are less salient in the PCP plot. Therefore, we expect users to be (**a**) slower, (**b**) less accurate, and (**c**) less confident.

H2. *Without clutter*, users perform better in a cluster identification task when the axes are ordered by `SIM` instead of `DIS` as clusters can be followed more easily. In particular, we expect users to be (**a**) faster, (**b**) more accurate, and (**c**) more confident with `SIM`.

H3. *With clutter*, users perform better in a cluster identification task when the axes are ordered by `DIS` instead of `SIM` as clusters are visually more prominent. In particular, we expect users to be (**a**) faster, (**b**) more accurate, and (**c**) more confident with `DIS`.

4.2. Benchmark Dataset and Ground Truth

To evaluate our hypotheses, benchmark datasets with ground truth information and increasing clutter levels are needed. We are not aware of such datasets for a cluster identification task. Therefore, we developed our own benchmark, consisting of ten popular real-world, and 72 synthetically created datasets along with the ground truth information. We make our dataset available in order to overcome the limitation of publicly available benchmark datasets [SNE*16] and to support the evaluation of PCP enhancements and reordering techniques in the future. For comparison, we also present a PCP with each dataset and reordering strategy in the supplementary material.

Synthetic datasets. We limit the dimensionality of all synthetic datasets to eight. This allows us to create complex cluster structures while keeping the expected time for the study in a reasonable time frame. We alternated the number of clusters between one and four and varied the structures of the clusters – ranging from linear clusters towards a high variance on all scales of the different axes.

Using the PCDC tool [BHvLF12], we manually created 24 *base datasets* ($\{1,2,3,4\}$ *clusters* $\times$ 6 *variations*) which fulfill the following properties: (i) clusters are clearly visible and separated from each other, (ii) there is only one clustering result per dataset, (iii) each cluster is present in all eight dimensions, and (iv) no outliers are added as they would distort the existing patterns [AA01]. In up to two dimensions, we merged two or more clusters such that participants need to investigate all dimensions to identify a cluster. To make the clusters comparable across datasets, we kept the cluster size constant with small randomization in the range of $45-50$ data points and vary the diameter of every cluster in each dimension randomly in the range $0.15-0.30$. All dimensions are normalized in $0.0-1.0$.

Next, we designed different clutter levels. Pomerenke et al. [PDK*19] show that random clutter (randomly and equally distributed records in all axes) produces visible patterns in PCPs which look similar to clusters (*Ghost clusters*). In order to not accidentally include 'fake patterns' in our dataset, but also be fair w.r.t. random clutter, we use a mixture of 30% random and 70% linear clutter (for every record: uniform and random distribution in one dimension $\pm 0 - 0.15$ in all other dimensions). Using several pilot experiments, this setting seemed to be complex enough, but also without any undesired patterns. For each base dataset, we created two copies with different clutter levels, one with 150 (`150N`), one with 300 data points (`300N`). We used the same clutter datasets for all base datasets to make them comparable and ensure we do not encode additional patterns in some of the datasets. After finalizing all 72 datasets (24 base datasets $\times \{$`0N`,`150N`,`300N`$\}$), we randomized the order of the records to remove potential effects in the drawing process.

Real-world datasets. We added ten frequently used datasets to see

the performance in real settings. We selected the datasets based on common usage in PCP reordering (i.e., by choosing datasets used in the techniques described in Table 1). Hence, the number of dimensions and records differ compared to synthetic datasets. Dimensions range between $4-13$, and the number of records between $32-515$. Examples are the `wine`, `mt-cars`, and `ecoli` `dataset`. If present, we removed categorical dimensions and outliers. We used Ward's method [WJ63] to retrieve a hierarchical clustering and a visual inspection to determine the clusters in the data.

4.3. Implementation

To compare the 'optimal' `SIM` and `DIS` layout, we used Ankerst et al.'s reordering algorithm [ABK98] with an exhaustive search to find the axes ordering. For the `SIM` layout, we used the Euclidean distance and minimized the sum of distances. For `DIS`, we used the same algorithm but maximized the distances. We pre-computed the orderings for all datasets in advance. To run the study, we developed a web application that is available at http://subspace.dbvis.de/pcp-study. The parallel coordinates plots have a size of 960×500 pixels and use color for the polylines to separate them from the axes which are colored in black. We did not add any design variations to the chart (i.e., transparency, or edge bundling) to avoid confounding factors.

4.4. Tasks and Data Randomization

Our study consisted of 21 trials per participant, which are grouped into three tasks that build on top of each other. The tasks were executed in increasing difficulty: Tasks 1, 2, and 3. Between two trials of a task, we showed a white screen with the term '*break time*', and participants had to click a button to continue with the next trial.

4.4.1. Task 1 (Similarity of Axes-Pairs)

We wanted to find out which *visual structures* support users in a cluster identification task. In particular, we were interested whether users find neighboring axes with a high similarity or dissimilarity more useful. In each trial, we showed the participants a PCP in which the number of clusters had to be counted. Users selected the number which they identified using four radio buttons (i.e., 1, 2, 3, 4) and a *can't tell* option. After the selection was confirmed, we showed a single radio button between each neighboring axes (see Figure 3 left) and asked the participants to select the pair which supported them best. Only one pair could be selected.

Randomization. We randomly picked three synthetic datasets with a different number of clusters. In the first trial, we showed `0N` clutter, in the second `150N`, and finally `300N` (increasing difficulty). As we are interested in whether participants prefer (dis-)similar neighboring axes, we arranged dimensions such that the PCP contains both similar and dissimilar axes pairs. To do so, we computed the similarity of neighboring axes using the Euclidean distance and used the maximal variance of similarities (`MaxVar`) as an objective function (see example in Figure 4). In summary:

3	levels of clutter (`0N`, `150N`, `300N`)	$\times$
31	participants	=
93	**trials in total**	

Post-processing. We collected the time to identify the number of clusters along with the similarity value of the selected axes pair. For comparison across datasets, we applied a linear min-max normalization to the similarity values of all neighboring axes pairs within each
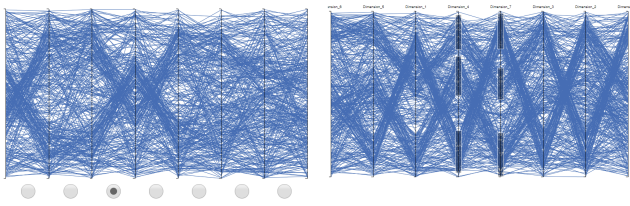
Figure 3: *User study interface for Task 1 (left) and Task 2 (right).*

(a) *no clutter (`0N`)*   **(b)** *150 clutter points (`150N`)*

Figure 4: *Dimensions are ordered by* MAXVAR *(maximizing the variance of similarities among neighboring axes). The result combines similar and dissimilar dimension pairs in one PCP.*

dataset. Pairs with the highest similarity are represented with 0.0, while high-dissimilarity pairs are represented by a value close to 1.0.

4.4.2. Task 2 (Cluster Identification and Selection)

We wanted to find out if participants are *better and more confident using a particular ordering strategy*. In each trial, we presented the participant one PCP, which was sorted by either SIM or DIS. The participant had to mark all clusters by choosing a pair of neighboring axes and marking every cluster in both axes using a brush feature (Figure 3 right). Brushing is applied by pressing the mouse button and marking the cluster along the axis. The selection can be moved, resized, or deleted. We do not highlight any data lines during or after brushing. After confirming their selections, participants rated their confidence in a correct clustering on a 5-point Likert scale.

Randomization. We selected 12 synthetic base datasets, three for each number of clusters, and randomized the order. Then, we distributed the datasets into three equal-sized groups: 0N, 150N, and 300N. Finally, we added four randomly selected real-world datasets in a new group RW. Within each group, we randomly applied twice a SIM and twice a DIS ordering. Participants worked on each group in order of increasing difficulty (i.e., clutter level). In summary:

4	levels of clutter (0N, 150N, 300N, RW)	×
2	repetitions	×
2	ordering strategies (SIM, DIS)	×
31	participants	=
496	**trials in total**	

Post-processing. We collected the time to mark the clusters and divided this by the number of clusters to be comparable across datasets. We also collected the selections and confidence levels.

We ignored clutter for the quality computation. We checked whether participants selected clusters in two neighboring dimensions and whether the number of clusters is therein consistent. In 132 trials, this was not the case, and we removed them from the data. The results are, however, still trustworthy as the removed trials are not skewed towards a particular reordering (66 trials each) or a clutter level (32, 28, 28, and 44 trials). For all correct trials, we then mapped the clusters between the selected axes together. First, we compute the Overlap coefficient [Szy34] between all cluster combinations and then merge the clusters with the highest overlap together. For each cluster combination, we keep the intersected set of data records as cluster members. The Overlap coefficient measures the overlap of members of the two clusters C_i and C_j: $overlap(C_i, C_j) = |C_i \cap C_j| / min(|C_i|, |C_j|)$.

The quality of an entire clustering is based on the Jaccard index [Jac01] between each selected cluster C_i and the corresponding ground truth cluster G_i. The Jaccard index measures the similarity between the clusters (record sets) C_i and G_i on a data record level:
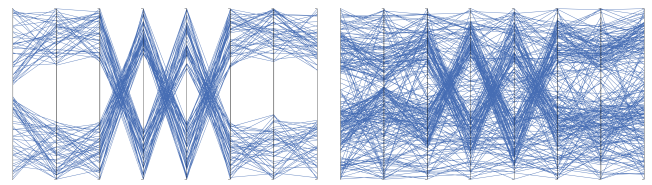
$jaccard(C_i, G_i) = |C_i \cap G_i| / |C_i \cup G_i|$. As participants can also select too few or too many clusters, our quality computation is a two-step process: First, we compute the average Jaccard index of each cluster to their best match in the ground truth. Second, we compute the average Jaccard index of every ground truth cluster to their best match of our selection. Our final clustering quality is then the average score of both steps.

4.4.3. Task 3 (Ordering Strategy Preferences)

We wanted to find out if participants have *preferences for a particular reordering* and why this is the case. We presented them two PCPs with the same dataset next to each other – one with SIM, one with DIS ordering. Using two radio buttons, participants had to select the preferred plot and then explain their choice in a free-text field.

Randomization. We randomly picked two synthetic datasets with a different number of clusters. In the first trial, we did not show any clutter (0N); in the second, we used either 150N or 300N (equally balanced across the participants). In the first trial, we used SIM ordering in the left, and DIS ordering in the right plot. In the second trial, we swapped the positions. In summary:

2	levels of clutter (0N, (150N ∨ 300N))	×
31	participants	=
62	**trials in total**	

Post-processing. We stored the preferred ordering and the text for each trial. Four participants reported not to see any preference between the options in one of the trials. We removed these participants from the statistical analysis, but report their choices in Section 5.3.

4.5. Participants and Procedure

Prior to the study, we conducted several pilot runs in order to determine appropriate clutter levels and the number of trials for each task.

Participants. To have participants with basic knowledge in information visualization and parallel coordinates, we conducted our user study during two lectures at the University of Konstanz, Germany. Both courses teach foundations in information visualization, one course for undergraduates, the other for graduates. The courses were taught by the same lecturer (not the authors), who also introduced and discussed the PCP technique two weeks prior to the study. We recruited 31 participants (17 male, 13 female, 1 NA). Their ages ranged from 19–31 years (median age 23). Each participant had finished high school, and 17 held a Bachelor's degree. The academic background was in the area of data analysis with 24 computer science, and 7 social and economic data analysis students. All participants reported having normal or corrected to normal vision.
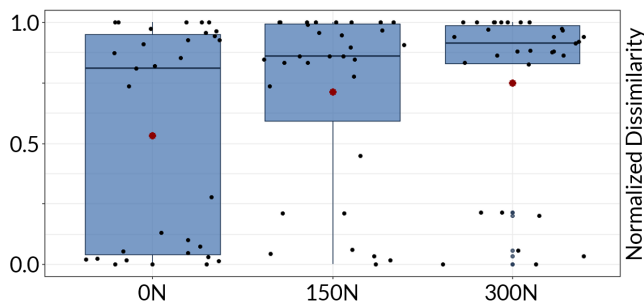
Figure 5: *Similarity of preferred neighboring dimensions (Task 1). With clutter, axes pairs with dissimilarity are preferred. Without clutter (`0N`), the preferences are almost equally balanced.*

Training and Procedure. All participants had to fill out a data privacy form in which we describe the data collected during the study. The participants sat scattered across the room and were not able to talk to each other. One of the authors started the study with a 30-minute recap on PCPs and comparing its visual patterns with scatter plots, discussing the advantages and disadvantages of the two techniques, and arguing about the effects of clutter, noise, and outliers. During the training, we did not provide strategies on how to identify clusters (or any other pattern) in PCP. Instead, we showed patterns in scatter plots and let all participants draw the respective patterns in a PCP (see training material). After this recap, we started the training. All participants opened their laptops and used a browser of their choice to access our online study. We provided a training platform, including all three tasks, but only two trials per task. We explained to the participants how to interact with the tool, and let the participants play around with the different trials. After answering the remaining questions, we made sure that all participants activated the full-screen mode within their browser and checked that the entire study could be conducted without scrolling for the different tasks. Participants needed between 20–30 minutes to complete the study.

5. User Study Results

We now report the summary statistics and highlight significant results ($p < .05$) in the data. For all tests, we checked the necessary preconditions, which can be found in the supplementary material along with the R scripts to reproduce the results. We used a one-sample Kolmogorov-Smirnov test to check if the data follows a normal distribution and Mauchly's test to check for sphericity.

5.1. Task 1 (Similarity of Axes-Pairs)

We used a repeated-measures ANOVA for the analysis of *completion time*. The post hoc analysis was done with a Bonferroni corrected t-test for dependent samples. As the *similarity of axes-pairs* was not normally distributed, a non-parametric Friedman's test was used.

Efficiency to Identify the Number of Clusters
There was a significant effect of *clutter* on *completion time* ($F(2,60) = 6.07$, $p < .01$, $\eta^2 = .10$). Post hoc comparisons revealed that *completion time* was significantly lower for the *clutter* condition `0N` ($\mu = 9.44s$) compared to `150N` ($p < .01$, $\mu = 16.67s$), and `300N` ($p < .01$, $\mu = 17.47s$), but not between `150N` and `300N` ($p = 1.0$).

Similarity of Selected Axes-Pair
No significant results can be reported ($\chi^2(2) = 4.77$, $p = .09$). As
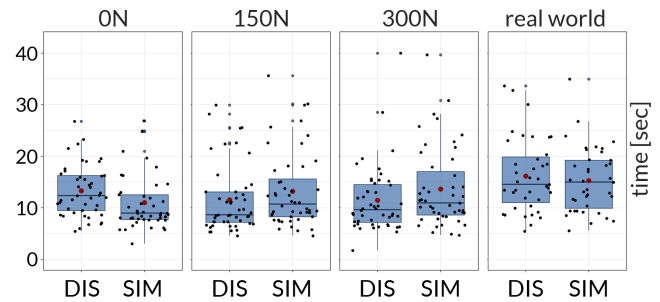

Figure 6: *Time to select clusters (Task 2).*

shown in Figure 5, the mean of the distances for the different *clutter conditions* were `0N` ($\mu = .53$), `150N` (.71), and `300N` (.75).

5.2. Task 2 (Cluster Identification and Selection)

For the comparison between *clutter* levels (independent of the ordering), we used a Kruskal-Wallis test, and a Bonferroni corrected Wilcoxon signed-rank test for post hoc analysis. For the *confidence*, we applied a Pearson's Chi-square test. To analyze the differences between the *ordering strategies* within each clutter level, we used a Wilcoxon signed-rank test for the analysis of *completion time*, *cluster quality*, and *confidence*. Data were split according to levels of *clutter* to compare the differences between `SIM` and `DIS`.

Efficiency to Identify and Mark Clusters
Between clutter levels, the medians of *completion time* for `0N`, `150N`, `300N`, and `RW` were 10.79, 9.53, 10.26, and 15.02, respectively. A Kruskal-Wallis test showed a significant effect on *clutter* level ($\chi^2(3) = 23.31$, $p < .001$). A post hoc test using Wilcoxon signed-rank tests showed only significant differences between `RW` and `0N`, `150N`, and `300N` (all $p < .01$).

As shown in Figure 6, the medians of the *completion time* for the `0N` *clutter* condition, for `SIM` and `DIS` were $8.9s$ and $12.35s$, respectively. A Wilcoxon signed-rank test showed that there was a significant effect of *ordering* strategy ($W = 1$, $Z = -2.46$, $p < .05$, $r = .25$). For the other *clutter* conditions, no significant results can be reported. For the `150N` *clutter* condition, the medians of *completion time* for `SIM` and `DIS` were $10.67s$ and $8.58s$, respectively ($p = .05$), for the `300N` *clutter* condition, the medians of *completion time* for `SIM` and `DIS` were $11.12s$ and $9.62s$, respectively ($p = .05$), and for the `RW` condition, the medians of *completion time* for `SIM` and `DIS` were $15.07s$ and $14.79s$, respectively ($p = .82$).

Quality of Identified and Marked Clusters
Between *clutter* levels, the medians of *quality* for `0N`, `150N`, `300N`, and `RW` were .98, .91, .85 and .37, respectively. A Kruskal-Wallis test showed a significant effect on *clutter* level ($\chi^2(3) = 181.56$, $p < .001$). A post hoc test using Wilcoxon Sign-rank tests showed the significant differences between `0N` and `150N` ($p < .001$), `300N` ($p < .001$), and `RW` ($p < .001$). Also, there were significant effects between `150N` and `300N` ($p < .05$), and `RW` ($p < .001$). Finally, `300N` and `RW` were also significantly different ($p < .001$).

The results of the cluster quality are summarized in Figure 7. For the `150N` *clutter* condition, the medians of *quality* for `SIM` and `DIS` were .87 and .94, respectively. A Wilcoxon signed-rank test showed a significant effect of *ordering* strategy ($W = 1$, $Z = -2.62$, $p < .001$, $r = .27$). For the `300N` *clutter* condition, the medians of
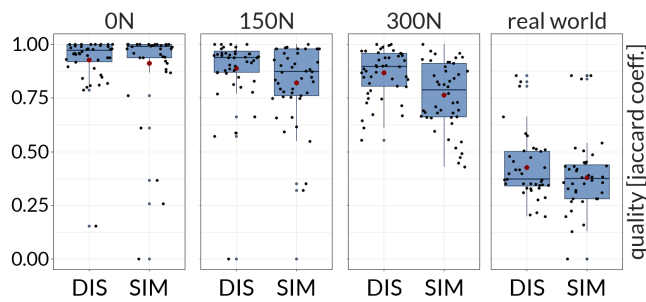
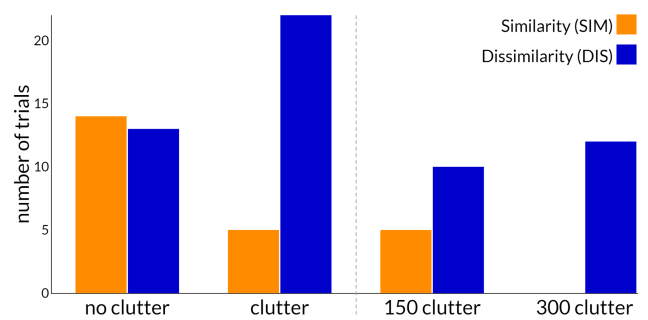**Figure 7:** *Quality of selected clusters (Task 2).*

Figure 8: *Preference of reordering strategy (Task 3). No preference for datasets without clutter. Participants strongly preferred a dissimilarity-based layout with an increasing amount of clutter. The figure shows both clutter levels combined ('clutter') and separately.*

quality for SIM and DIS were .79 and .90, respectively. A Wilcoxon signed-rank test showed a significant effect of *ordering strategy* ($W = 1, Z = -3.36, p < .001, r = .34$). The other levels of *clutter* did not show a significant difference. The medians of the quality score in the ON *clutter* condition were .99 for SIM and .97 for DIS ($p = .33$), and in the RW condition .37 for both SIM and DIS ($p = .25$).

Confidence of Marked Clusters
An overview of the participants' confidence is shown in Figure 9. Between *clutter levels*, the medians of *confidence* for ON, 150N, 300N, and RW were 2, 1, 1, and 0, respectively. A Pearson Chi-square test showed a significant effect of *clutter* level on *confidence* ($\chi^2(12) = 120.97, p < .001$). Post hoc analysis revealed significant differences between ON and 150N ($p < .001$), 300N ($p < .001$), and RW ($p < .001$); between 150N and 300N ($p < .05$), and RW ($p < .001$); and between 300N and RW ($p < .005$).

For the 150N *clutter* condition, the medians of *confidence* for SIM and DIS were both 1. A Wilcoxon signed-rank test showed a significant effect of *ordering* strategy ($W = 1, Z = -2.52, p < .05, r = .26$). For the 300N *clutter* condition, the medians of *confidence* for SIM and DIS were 0 and 1, respectively. A Wilcoxon signed-rank test showed a significant effect of *ordering* strategy ($W = 1, Z = -3.75, p < .001, r = .38$). The remaining levels of *clutter* did not show a significant difference between *ordering* strategies with the same medians for SIM and DIS (ON = 2, $p = .25$; RW = 0, $p = .48$).

5.3. Task 3 (Understanding Preferences)

The distribution of preferences is shown in Figure 8. Two participants selected SIM, ten participants DIS for both *clutter* conditions. Twelve participants preferred SIM without *clutter* and changed their preference to DIS for the second trial, which included *clutter*. Vice versa, three participants changed from DIS to SIM. A binomial test showed a significant difference ($p < .05$) in the proportion of *preference* ($\chi^2(1, N = 27) = 12$). The probability of success was .8.

Three out of four of the removed participants (see Section 4.4.3) did not have a preference for ON, but preferred DIS for cluttered datasets. One participant preferred SIM for clutter-free datasets and had no preference for the dataset with clutter.

6. Discussion

We discuss the participants' performance according to our hypotheses and highlight findings from the qualitative feedback of task 3.

Influence of Clutter (H1). Increasing the amount of clutter has a negative effect on the *quality* of the cluster identification and *confidence* of participants. These findings confirm **H1 (b)** and **(c)**. While

we see an increasing *completion time* for higher clutter levels in task 1, we cannot verify this finding in the second task. Hence, we cannot make a final judgment on **H1 (a)**. As expected, clutter negatively influences the cluster identification. Patterns may vanish due to overlapping data lines, making the identification more difficult. Therefore, visualization experts need to carefully design PCPs and reduce the amount of clutter if possible (e.g., sampling).

Reordering for Clutter-free Datasets (H2). Similarity-based ordering strategies are a good choice for datasets without clutter. Our results show that participants perform the identification of clusters more efficiently when working with a SIM layout, which confirms **H2 (a)**. It seems as if participants are faster in combining straight data lines into clusters in contrast to data lines with strong slopes. A possible explanation could be the Gestalt law [Wer23, War20] of *continuation*, which could help participants in tracking data lines across dimensions. Kellman and Shipley [KS91] support this argument: the angular parameters, determining the grouping of lines to clusters, may support the ability to find clusters across multiple sets of axes. The qualitative feedback also confirms our findings. Participants reported, for example, that *"The structure of clusters is clearer"*, *"[clusters] don't cross very often"*, or they prefer SIM *"[...] since they do not intersect with each other in the majority of areas between each two dimensions [...]"*. We cannot support hypotheses **H2 (b)** and **(c)**. There are no significant differences in the cluster identification *quality* or the *confidence* of the participants (see also Figure 7 and 9). Also, participants did not have a subjective preference for a particular reordering strategy, as shown in Figure 8.

Reordering for Cluttered Datasets (H3). For datasets with clutter, there is strong evidence that DIS layout strategies are more suitable. The *quality* of marked clusters is significantly better when participants used a DIS ordering strategy, confirming **H3 (b)**. This finding coincides with the reported *confidence* of participants (see Figure 9), who are also significantly more confident when working with DIS in clutter conditions. Even if both options are available (SIM and DIS), there is statistical proof that participants will choose a DIS ordering in clutter conditions providing evidence for **H3 (c)**. The Gestalt law of *grouping by orientation similarly* [Wer23, War20] might be a reason for this preference. The orientation of the lines is more salient in the DIS ordering, which facilitates a stronger grouping compared to a SIM ordering.

We cannot see significant differences in the similarity values of
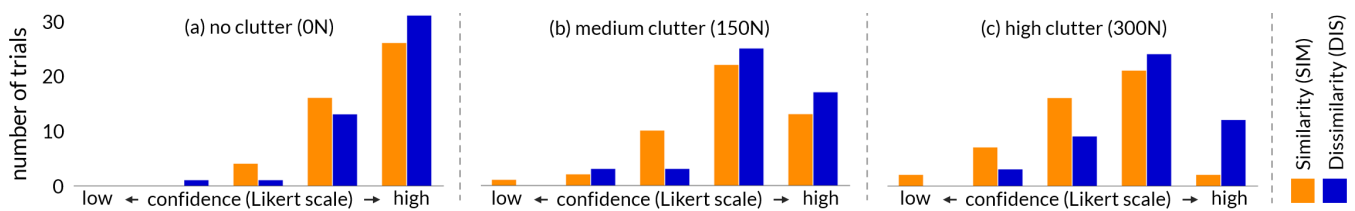
Figure 9: *Confidence of marked clusters (Task 2). Participants have higher confidence in their cluster selection using dissimilarity ordering.*

the selected axes pairs in task 1. However, Figure 5 illustrates the distribution of similarity values, providing evidence that participants believe that `DIS` axes pairs support them better in a cluster identification task. These findings are also in line with the qualitative feedback. The majority stated a preference for `DIS` over a `SIM` layout (see Figure 8). The participants said, for example, *"the spikes make the clusters more obvious"*, *"clearer [in A] because the zig-zag makes it easier to see among the noise."*, or *"The lines are closer together"*.

As shown in Figure 6, participants performed the cluster identification task faster with a `DIS` layout compared to a `SIM` layout in cluttered datasets. However, the differences in the *completion time* are not significant ($p = 0.05$). Therefore, we cannot confirm **H3 (a)**.

6.1. Design Considerations

With the results gained from our study, we derive the following design considerations for using PCPs in a cluster identification task.

Whenever possible, clutter should be removed in a pre-processing step. Results from tasks 1 and 2 indicate that participants working with PCPs need more time, are less accurate, and are less confident in identifying clusters with an increasing amount of clutter.

For datasets without any clutter, a `SIM` layout should be preferred over a `DIS` layout. Participants working with a `SIM` ordering strategy were faster in identifying and marking clusters compared to a `DIS` layout, as results indicate in task 2. There is, however, no difference in the quality of the clustering or confidence.

When clutter is an issue, a `DIS` ordering strategy should be preferred over a `SIM` layout. As we can see in the results from tasks 2 and 3, participants performed more accurately and were more confident in their selection when working with a `DIS` layout.

Although we used two clutter conditions in our study, it is challenging to derive specific guidelines when a dataset is considered as *cluttered*. It depends on many properties, such as the number of records, the general density of data and patterns, and the size of the PCP. Therefore, this needs to be analyzed in follow-up studies.

6.2. Limitations and Future Work

In our study, we focused on cluster identification. Therefore, the proposed design considerations need to be considered with caution for other tasks like correlation analysis. There might be changes in performance due to different patterns of interest. The same is true for the cluster structures. Our synthetic benchmark consists of a strong cluster structure throughout all dimensions. The results of our study might not be representative if cluster structures are less compact or not present across all dimensions. The results of our study already show that real-world datasets perform significantly worse than the synthetic benchmark data, although we selected datasets, commonly used in PCP research. The reasons for this effect might be that (1) all

records of the real-world data belong to a cluster (no clutter was present), (2) the clusters in the real-world dataset were less compact than in the synthetic datasets, and (3) the real-world trials were done right after the `300N` trials. Hence, participants selected only a subset to be part of a cluster and interpreting the remaining points as clutter. To generalize our results further, a follow-up study with different datasets should be conducted.

Further limitations of the study are (1) **The number of dimensions and records:** while we believe that results are independent of the number of dimensions, we restricted ourselves to eight dimensions to keep the trials throughout the study comparable. With an increasing number of dimensions, the computation of ordering algorithms will take longer; however, this was out of the scope for our study to investigate. (2) **The population of participants:** The study was conducted during two InfoVis lectures at our university. Therefore, participants were recruited from an InfoVis-trained, local student population, limiting the generalizability of the results. (3) **Study setup:** Participants used their own laptops with different screen space and resolution. Although we manually checked that the study was displayed correctly on each laptop, the experience might change due to different screen settings or selected internet browser.

7. Conclusion

This paper advances the field of axes reordering in parallel coordinates plots (PCPs). First, we classified existing reordering techniques based on their inner workings, preferred patterns, and meta characteristics. Using this classification, we provide guidance in selecting an appropriate approach for a given task. Second, we pushed the evaluation of axes reordering techniques towards empirical justification. We conducted the first controlled user study to assess the performance of PCPs with two different ordering strategies. Specifically, we investigated whether the often proposed similarity-based axes arrangement (`SIM`) is better to identify clusters than a dissimilarity-based layout (`DIS`), which produces more salient cluster patterns. Our results show that, depending on the clutter level, participants performed differently based on the used ordering strategy. When no clutter was present, a `SIM` layout was more efficient, whereas, for cluttered datasets, a `DIS` layout led to better results. The subjective preference of participants supported these findings. Thus, our study shows that the performance of participants can be increased by choosing the correct layout strategy based on the underlying task.

Acknowledgments

References

[AA01] ANDRIENKO G., ANDRIENKO N.: Constructing parallel coordinates plot for problem solving. In *International Symposium on Smart Graphics* (2001), pp. 9–14. 1, 7

[ABK98] ANKERST M., BERCHTOLD S., KEIM D. A.: Similarity Clustering of Dimensions for an Enhanced Visualization of Multidimensional Data. In *IEEE Symposium on Information Visualization* (1998), pp. 52–60. doi:10.1109/INFVIS.1998.729559. 1, 2, 4, 5, 7

[AdOL06] ARTERO A. O., DE OLIVEIRA M. C. F., LEVKOWITZ H.: Enhanced High Dimensional Data Visualization through Dimension Reduction and Attribute Arrangement. In *International Conference on Information Visualisation* (2006), pp. 707–712. doi:10.1109/IV.2006.49. 2, 4, 5

[AKSZ13] ACHTERT E., KRIEGEL H., SCHUBERT E., ZIMEK A.: Interactive data mining with 3D-parallel-coordinate-trees. In *Proceedings of the International Conference on Management of Data* (2013), pp. 1009–1012. doi:10.1145/2463676.2463696. 3

[AZZ10] ALSAKRAN J., ZHAO Y., ZHAO X.: Tile-based parallel coordinates and its application in financial visualization. In *Visualization and Data Analysis* (2010), p. 753003. doi:10.1117/12.838819. 1

[BBK*18] BEHRISCH M., BLUMENSCHEIN M., KIM N. W., SHAO L., EL-ASSADY M., FUCHS J., SEEBACHER D., DIEHL A., BRANDES U., PFISTER H., SCHRECK T., WEISKOPF D., KEIM D. A.: Quality Metrics for Information Visualization. *Comput. Graph. Forum 37*, 3 (2018), 625–662. doi:10.1111/cgf.13446. 2, 3

[BHvLF12] BREMM S., HESS M., VON LANDESBERGER T., FELLNER D. W.: PCDC - On the Highway to Data - A Tool for the Fast Generation of Large Synthetic Data Sets. In *EuroVis Workshop on Visual Analytics* (2012). doi:10.2312/PE/EuroVAST/EuroVA12/007-011. 7

[BTK11] BERTINI E., TATU A., KEIM D.: Quality Metrics in High-Dimensional Data Visualization: An Overview and Systematization. *IEEE Trans. Vis. Comput. Graph. 17*, 12 (2011), 2203–2212. doi:10.1109/TVCG.2011.229. 2

[DK10] DASGUPTA A., KOSARA R.: Pargnostics: Screen-Space Metrics for Parallel Coordinates. *IEEE Trans. Vis. Comput. Graph. 16*, 6 (2010), 1017–1026. doi:10.1109/TVCG.2010.184. 1, 2, 3, 4, 5

[ED06] ELLIS G. P., DIX A. J.: Enabling Automatic Clutter Reduction in Parallel Coordinate Plots. *IEEE Trans. Vis. Comput. Graph. 12*, 5 (2006), 717–724. doi:10.1109/TVCG.2006.138. 1, 2

[ED07] ELLIS G. P., DIX A. J.: A Taxonomy of Clutter Reduction for Information Visualisation. *IEEE Trans. Vis. Comput. Graph. 13*, 6 (2007), 1216–1223. doi:10.1109/TVCG.2007.70535. 2

[FBT*10] FERDOSI B. J., BUDDELMEIJER H., TRAGER S. C., WILKINSON M. H. F., ROERDINK J. B. T. M.: Finding and visualizing relevant subspaces for clustering high-dimensional astronomical data using connected morphological operators. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology* (2010), pp. 35–42. doi:10.1109/VAST.2010.5652450. 6

[FR11] FERDOSI B. J., ROERDINK J. B. T. M.: Visualizing High-Dimensional Structures by Dimension Ordering and Filtering using Subspace Analysis. *Comput. Graph. Forum 30*, 3 (2011), 1121–1130. doi:10.1111/j.1467-8659.2011.01961.x. 2, 4, 5, 6

[FWR99] FUA Y.-H., WARD M. O., RUNDENSTEINER E. A.: Hierarchical Parallel Coordinates for Exploration of Large Datasets. In *Proceedings of the IEEE Conference on Visualization* (1999), pp. 43–50. doi:10.1109/VISUAL.1999.809866. 1, 2

[HGM*97] HOFFMAN P., GRINSTEIN G. G., MARX K. A., GROSSE I., STANLEY E.: DNA visual and analytic data mining. In *IEEE Visualization* (1997), pp. 437–442. doi:10.1109/VISUAL.1997.663916. 1

[HHJ11] HUANG T., HUANG M. L., JIN J. S.: Parallel Rough Set: Dimensionality Reduction and Feature Discovery of Multi-Dimensional Data in Visualization. In *Proceedings of International Conference on Neural Information Processing* (2011), pp. 99–108. doi:10.1007/978-3-642-24958-7_12. 4, 5

[HKP11] HAN J., KAMBER M., PEI J.: *Data Mining: Concepts and Techniques, 3rd edition*. Morgan Kaufmann, 2011. URL: http://hanj.cs.illinois.edu/bk3/. 3

[HO10] HURLEY C. B., OLDFORD R.: Pairwise display of high-dimensional information via eulerian tours and hamiltonian decompositions. *Journal of Computational and Graphical Statistics 19*, 4 (2010), 861–886. URL: https://www.jstor.org/stable/25765378. 2

[Hou62] HOUGH P. V.: Method and means for recognizing complex patterns, 1962. US Patent 3,069,654. 5

[HvW10] HOLTEN D., VAN WIJK J. J.: Evaluation of Cluster Identification Performance for Different PCP Variants. *Comput. Graph. Forum 29*, 3 (2010), 793–802. doi:10.1111/j.1467-8659.2009.01666.x. 2

[HW13] HEINRICH J., WEISKOPF D.: State of the Art of Parallel Coordinates. In *Eurographics* (2013), pp. 95–116. doi:10.2312/conf/EG2013/stars/095-116. 2, 3

[HW15] HEINRICH J., WEISKOPF D.: Parallel Coordinates for Multidimensional Data Visualization: Basic Concepts. *Computing in Science and Engineering 17*, 3 (2015), 70–76. doi:10.1109/MCSE.2015.55. 3

[HYFC14] HARRISON L., YANG F., FRANCONERI S., CHANG R.: Ranking Visualizations of Correlation Using Weber's Law. *IEEE Trans. Vis. Comput. Graph. 20*, 12 (2014), 1943–1952. doi:10.1109/TVCG.2014.2346979. 1

[Ins85] INSELBERG A.: The plane with parallel coordinates. *The Visual Computer 1*, 2 (1985), 69–91. doi:10.1007/BF01898350. 1

[Ins09a] INSELBERG A.: *Data Mining and Other Applications*. Springer, 2009, pp. 379–427. doi:10.1007/978-0-387-68628-8_10. 2

[Ins09b] INSELBERG A.: *Parallel Coordinates*. Springer, 2009, pp. 2018–2024. doi:10.1007/978-0-387-39940-9_262. 1

[Jac01] JACCARD P.: Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bulletin de la Société vaudoise des sciences naturelles 37* (1901), 547–579. 8

[JF16] JOHANSSON J., FORSELL C.: Evaluation of Parallel Coordinates: Overview, Categorization and Guidelines for Future Research. *IEEE Trans. Vis. Comput. Graph. 22*, 1 (2016), 579–588. doi:10.1109/TVCG.2015.2466992. 1, 2, 3

[JFLC08] JOHANSSON J., FORSELL C., LIND M., COOPER M. D.: Perceiving patterns in parallel coordinates: determining thresholds for identification of relationships. *Information Visualization 7*, 2 (2008), 152–162. doi:10.1057/palgrave.ivs.9500166. 2

[JHB*17] JÄCKLE D., HUND M., BEHRISCH M., KEIM D. A., SCHRECK T.: Pattern Trails: Visual Analysis of Pattern Transitions in Subspaces. In *IEEE Conference on Visual Analytics Science and Technology* (2017), pp. 1–12. doi:10.1109/VAST.2017.8585613. 3

[JJ09] JOHANSSON S., JOHANSSON J.: Interactive Dimensionality Reduction Through User-defined Combinations of Quality Metrics. *IEEE Trans. Vis. Comput. Graph. 15*, 6 (2009), 993–1000. doi:10.1109/TVCG.2009.153. 1, 2, 3, 4, 5, 6

[JLJC05] JOHANSSON J., LJUNG P., JERN M., COOPER M.: Revealing structure within clustered parallel coordinates displays. In *IEEE Symposium on Information Visualization* (2005), pp. 125–132. doi:10.1109/INFVIS.2005.1532138. 1, 2

[KAC15] KANJANABOSE R., ABDUL-RAHMAN A., CHEN M.: A Multitask Comparative Study on Scatter Plots and Parallel Coordinates Plots. *Comput. Graph. Forum 34*, 3 (2015), 261–270. doi:10.1111/cgf.12638. 1, 2

[KS91] KELLMAN P. J., SHIPLEY T. F.: A theory of visual interpolation in object perception. *Cognitive Psychology 23*, 2 (1991), 141 – 221. doi:10.1016/0010-0285(91)90009-D. 10

[KZZM12] KUANG X., ZHANG H., ZHAO S., MCGUFFIN M. J.: Tracing Tuples Across Dimensions: A Comparison of Scatterplots and Parallel Coordinate Plots. *Comput. Graph. Forum 31*, 3 (2012), 1365–1374. doi:10.1111/j.1467-8659.2012.03129.x. 1

[LHH12] LU L. F., HUANG M. L., HUANG T.: A New Axes Re-ordering Method in Parallel Coordinates Visualization. In *International Conference on Machine Learning and Applications* (2012), pp. 252–257. doi:10.1109/ICMLA.2012.148. 4, 5

[LHZ16] LU L. F., HUANG M. L., ZHANG J.: Two axes re-ordering methods in parallel coordinates plots. *J. Vis. Lang. Comput. 33* (2016), 3–12. doi:10.1016/j.jvlc.2015.12.001. 3, 4, 6

[LMvW10] LI J., MARTENS J., VAN WIJK J. J.: Judging correlation from scatterplots and parallel coordinate plots. *Information Visualization 9*, 1 (2010), 13–30. doi:10.1057/ivs.2008.13. 1

[MM08] MCDONNELL K. T., MUELLER K.: Illustrative Parallel Coordinates. *Computer Graphics Forum 27*, 3 (2008), 1031–1038. doi:10.1111/j.1467-8659.2008.01239.x. 1

[Mou11] MOUSTAFA R. E.: Parallel coordinate and parallel coordinate density plots. *WIREs Computational Statistics 3*, 2 (2011), 134–148. doi:10.1002/wics.145. 2

[MTJ12] MAKWANA H., TANWANI S., JAIN S.: Axes Re-Ordering in Parallel Coordinate for Pattern Optimization. *International Journal of Computer Applications 40*, 13 (2012), 43–48. doi:10.5120/5044-7370. 4, 5

[MW95] MARTIN A. R., WARD M. O.: High Dimensional Brushing for Interactive Exploration of Multivariate Data. In *Proceedings of the Conference on Visualization* (1995), pp. 271–278. doi:10.1109/VISUAL.1995.485139. 1, 2

[NGC01] NAGESH H. S., GOIL S., CHOUDHARY A. N.: Adaptive Grids for Clustering Massive Data Sets. In *Proceedings of the International Conference on Data Mining* (2001), pp. 1–17. doi:10.1137/1.9781611972719.7. 6

[NVE*17] NETZEL R., VUONG J., ENGELKE U., O'DONOGHUE S. I., WEISKOPF D., HEINRICH J.: Comparative eye-tracking evaluation of scatterplots and parallel coordinates. *Visual Informatics 1*, 2 (2017), 118–131. doi:10.1016/j.visinf.2017.11.001. 5, 6

[Paw12] PAWLAK Z.: *Rough sets: Theoretical aspects of reasoning about data*, vol. 9. Springer Science & Business Media, 2012. 5

[PDK*19] POMERENKE D., DENNIG F. L., KEIM D. A., FUCHS J., BLUMENSCHEIN M.: Slope-Dependent Rendering of Parallel Coordinates to Reduce Density Distortion and Ghost Clusters. In *IEEE Visualization Conference (VIS)* (2019), pp. 86–90. doi:10.1109/VISUAL.2019.8933706. 2, 6, 7

[PL17] PELTONEN J., LIN Z.: Parallel Coordinate Plots for Neighbor Retrieval. In *Proceedings of the International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications* (2017), pp. 40–51. doi:10.5220/0006097400400051. 1, 2, 3, 4, 5, 6

[PWR04] PENG W., WARD M. O., RUNDENSTEINER E. A.: Clutter Reduction in Multi-Dimensional Data Visualization Using Dimension Reordering. In *Symposium on Information Visualization* (2004), pp. 89–96. doi:10.1109/INFVIS.2004.15. 1, 2, 4, 5

[SC*05] STOCKINGER K., , CAMPBELL S., LAU S., FISK M., GAVRILOV E., KENT A., DAVIS C. E., OLINGER R., YOUNG R., PREWETT J., WEBER P., CAUDELL T. P., BETHEL E. W., SMITH S.: Network Traffic Analysis With Query Driven Visualization SC 2005 HPC Analytics Results. In *Proceedings of the ACM/IEEE Conference on Supercomputing* (2005), pp. 72–72. doi:10.1109/SC.2005.47. 1

[SCC*18] SCHLOERKE B., CROWLEY J., COOK D., HOFMANN H., WICKHAM H., BRIATTE F., MARBACH M., THOEN E., ELBERG A., LARMARANGE J.: *Ggally: Extension to ggplot2*, 2018. R package version 1.4.0. URL: https://cran.r-project.org/package=GGally. 3, 4, 6

[SKKAJ15] SUNIL KUMAR S., KRISHNAN T., ASHOK S., JUDY M.: Clutter Reduction in Parallel Coordinates using Binning Approach for Improved Visualization. *International Journal of Electrical & Computer Engineering (2088-8708) 5*, 6 (2015). doi:10.11591/ijece.v5i6.pp1564-1568. 1

[SNE*16] SCHULZ C., NOCAJ A., EL-ASSADY M., FREY S., HLAWATSCH M., HUND M., KARCH G. K., NETZEL R., SCHÄTZLE C., BUTT M., KEIM D. A., ERTL T., BRANDES U., WEISKOPF D.: Generative Data Models for Validation and Evaluation of Visualization Techniques. In *Proceedings of the Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization* (2016), pp. 112–124. doi:10.1145/2993901.2993907. 7

[Szy34] SZYMKIEWICZ D.: Une contribution statistique a la géographie floristique. *Polskie Towarzystwo Botaniczne* (1934). 8

[TAE*09] TATU A., ALBUQUERQUE G., EISEMANN M., SCHNEIDEWIND J., THEISEL H., MAGNOR M. A., KEIM D. A.: Combining automated analysis and visualization techniques for effective exploration of high-dimensional data. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology* (2009), pp. 59–66. doi:10.1109/VAST.2009.5332628. 5

[TAE*11] TATU A., ALBUQUERQUE G., EISEMANN M., BAK P., THEISEL H., MAGNOR M. A., KEIM D. A.: Automated Analytical Methods to Support Visual Exploration of High-Dimensional Data. *IEEE Trans. Vis. Comput. Graph. 17*, 5 (2011), 584–597. doi:10.1109/TVCG.2010.242. 2, 4, 5

[Tuf01] TUFTE E. R.: *The Visual Display of Quantitative Information*, 2nd ed. Graphics Press, Cheshire, CT, USA, 2001. 6

[Van15] VAN LONG T.: A new metric on parallel coordinates and its application for high-dimensional data visualization. In *IEEE International Conference on Advanced Technologies for Communications* (2015), pp. 297–301. doi:10.1109/ATC.2015.7388338. 4, 5

[WAG05] WILKINSON L., ANAND A., GROSSMAN R. L.: Graph-Theoretic Scagnostics. In *IEEE Symposium on Information Visualization* (2005), pp. 157–164. doi:10.1109/INFVIS.2005.1532142. 6

[War20] WARE C.: *Information visualization: perception for design*. Morgan Kaufmann, 2020. 10

[Weg90] WEGMAN E. J.: Hyperdimensional Data Analysis Using Parallel Coordinates. *Journal of the American Statistical Association 85*, 411 (1990), 664–675. doi:10.1080/01621459.1990.10474926. 3

[Wer23] WERTHEIMER M.: Untersuchungen zur Lehre von der Gestalt. II. *Psychologische Forschung 4*, 1 (1923), 301–350. doi:10.1007/BF00410640. 10

[WJ63] WARD JR J. H.: Hierarchical grouping to optimize an objective function. *Journal of the American statistical association 58*, 301 (1963), 236–244. URL: http://www.jstor.org/stable/2282967. 7

[XFJ*12] XIANG Y., FUHRY D., JIN R., ZHAO Y., HUANG K.: Visualizing Clusters in Parallel Coordinates for Visual Knowledge Discovery. In *Advances in Knowledge Discovery and Data Mining* (2012), Springer, pp. 505–516. doi:10.1007/978-3-642-30217-6_42. 4, 5

[YPWR03] YANG J., PENG W., WARD M. O., RUNDENSTEINER E. A.: Interactive Hierarchical Dimension Ordering, Spacing and Filtering for Exploration of High Dimensional Datasets. In *IEEE Symposium on Information Visualization* (2003), pp. 105–112. doi:10.1109/INFVIS.2003.1249015. 1, 2, 3, 4, 5, 6

[ZYQ*08] ZHOU H., YUAN X., QU H., CUI W., CHEN B.: Visual Clustering in Parallel Coordinates. *Comput. Graph. Forum 27*, 3 (2008), 1047–1054. doi:10.1111/j.1467-8659.2008.01241.x. 3

[ZYYC18] ZHOU Z., YE Z., YU J., CHEN W.: Cluster-aware arrangement of the parallel coordinate plots. *J. Vis. Lang. Comput. 46* (2018), 43–52. doi:10.1016/j.jvlc.2017.10.003. 4, 5