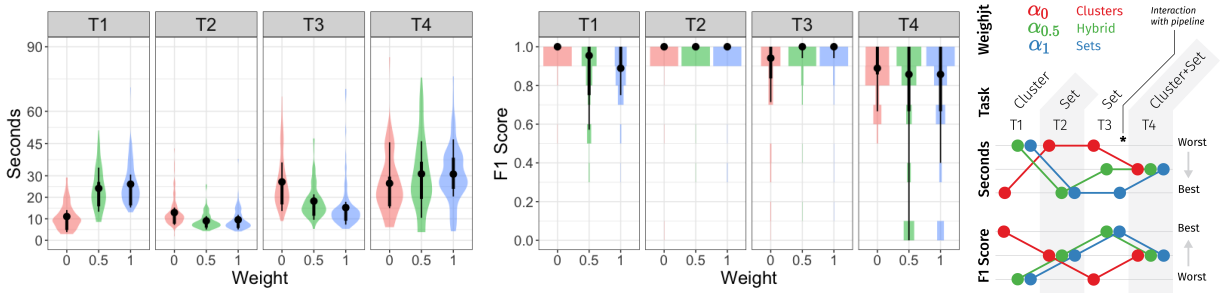
On Combined Visual Cluster and Set Analysis

Nikolaus Piccolotto* Markus Wallinger Silvia Miksch Markus Bögl

TU Wien



(a) Violin plots for T1–T4 time (lower is better).

(b) Histograms for T1–T4 accuracy (higher is better).

(c) Order relationships in a, b.

Figure 1: Response time (a) and accuracy (b) results for UT weights α_0, $\alpha_{0.5}$, and α_1, and their order (c). The dot marks mean (a) and median (b), while lines show 50%/75% highest density continuous interval (a) and 50/75 quantile interval (b).

Abstract

Real-world datasets often consist of quantitative and categorical variables. The analyst needs to focus on either kind separately or both jointly. We proposed a visualization technique tackling these challenges that supports visual cluster and set analysis. In this paper, we investigate how its visualization parameters affect the accuracy and speed of cluster and set analysis tasks in a controlled experiment. Our findings show that, with the proper settings, our visualization can support both task types well. However, we did not find settings suitable for the joint task, which provides opportunities for future research.

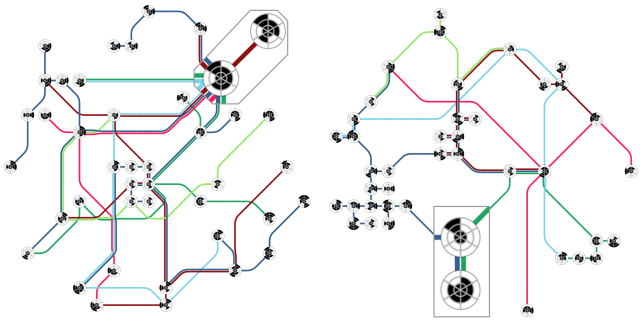
Index Terms: Visual cluster analysis, set visualization.

1 Introduction

Real-world datasets often consist of a mix of quantitative and categorical variables. Analysis tasks may alternatively or simultaneously involve i) cluster analysis based on multidimensional similarity and ii) set tasks on categorical variables. In previous work [16], we developed a visualization technique that tackles both tasks. It is called "UnDRground Tubes" (short UT) and based on a metro map metaphor. Multidimensional data points are visualized as glyphs and embedded in 2D using Multidimensional Scaling [18]. Overlaps are avoided by displacing glyphs onto regular grid positions. Users can control the glyph layout with UT 's *weight* parameter α. It defines how the data points' pairwise multidimensional and set distances are preserved. At $\alpha = 0$, distances in the grid correspond only to the former (Figure 2a), while at $\alpha = 1$, to the latter (Figure 2b). Settings in between will show a mix of set and multidimensional distances. Sets are visualized by colored lines that connect glyphs in the same sets. UT 's *pipeline* parameter controls whether lines are routed heuristically or optimally. Thus, users can flexibly switch between cluster and set analysis using UT. We extensively evaluated UT with qualitative approaches: An ICE-T [22] evaluation yielded very high scores, and experts successfully applied multidimensional and set tasks toward their analysis goals [16]. However, it remained an open question how well UT is suited to particular tasks when "opposite" weights are used, e.g., set tasks at $\alpha = 0$. Likewise, it was unknown what advantage, if any, weights $0 < \alpha < 1$ afford. Thus, we still have to devise UT guidelines. To do so, we describe a controlled experiment to answer the following research question: *How do accuracy and speed of cluster and set tasks relate to UT 's weight and pipeline parameters?*

Our contributions are thus the following:

- We devise an experiment to test the suitability of various UT visualization styles on cluster and set analysis tasks (Section 3).
- We analyze the experiment results using appropriate frequentist statistics (Section 4).
- Based on the analyzed results, we suggest guidelines for employing UT (Section 4.4).



(a) α_0 weight, optimal pipeline. Zoomed glyphs added as inset for better visibility.

(b) α_1 weight, heuristic pipeline. Zoomed glyphs added as inset for better visibility.

Figure 2: Example stimuli for task T3 (Section 3.1) from the experiment.

2 Related Work

Our work relates to empirical studies in the context of visual cluster analysis and set visualization.

2.1 Visual Cluster Analysis Using DR Scatterplots

Clustering is an essential wide-spread class of data analysis methods, and various flavors were proposed over time [26]. We focus on visual cluster analysis using 2D dimension reduction (DR) scatterplots

*e-mail:{firstname}.{lastname}@tuwien.ac.at

(multidimensional projections) [15]. Lewis and de Sa [10] investigated whether people judge DR scatterplot quality consistently and found only experts do, but novices do not. Sedlmair et al. [19] provide guidance on visual encoding choices of DR scatterplots regarding class separability. They found that 2D scatterplots are "mostly good enough," scatterplot matrices are sometimes useful, and interactive 3D approaches rarely. Based on the same datasets, Sedlmair et al. [20] also proposed a taxonomy of visual cluster separation factors. Tatu et al. [21] assessed whether some class separation metrics align with human judgment and found two metrics where this was the case. On the other hand, Etemadpour et al. [7] conducted a perception-based study. Using five projection methods, they compared how users could solve several cluster analysis tasks on four datasets, such as ranking clusters by density. As expected, no projection method outperformed the others in all tasks or datasets. Xia et al. [25] later conducted a similar experiment, but instead of individual techniques, they investigated the differences between local/global and linear/non-linear DR methods on cluster analysis tasks. No clear picture emerged here, either, as at least one outlier existed for each method and task. Espadoto et al. [6] began to tackle the issue that many projection methods require parameters that influence the projection quality. They set up a computational experiment and found, e.g., that UMAP with default parameters is a solid choice across datasets. Moriaru et al. [14] investigated which quality metrics can predict user preference of a DR scatterplot. They found that, e.g., scagnostics [24] or separability measures can be useful for that purpose. While their combination is starting to be explored [5], no experiment has considered DR projections in conjunction with categorical data and set tasks.

2.2 Set Visualization

Several works in the literature compare the effectiveness of set visualization idioms [2] across tasks. Some studies considered **fixed embeddings**, where the position of set members is fixed, such as in spatial sets. Alper et al. [1] compared LineSets to Bubble Sets. Tasks included identifying members of a set and judging cardinalities of set relations. Answers were more accurate with LineSets. Meulemans et al. [13] compared their proposed KelpFusion to LineSets and Bubble Sets. Tasks pertained mostly to the cardinality of sets or set intersections. They found that KelpFusion and LineSets are more accurate than Bubble Sets, while KelpFusion had faster answer times than LineSets. Rodgers et al. [17] compared visualization approaches for "grouped network data": Bubble Sets, LineSets, KelpFusion, EulerView, and their own proposal SetNet. Information about the network and set data were required to solve tasks. They concluded that EulerView and SetNet outperformed Bubble Sets, KelpFusion, and LineSets. Other studies focused on **free embeddings**. Chapman et al. [3] compared the effectiveness of Euler diagrams, two kinds of Venn diagrams, and linear diagrams for the set relation tasks intersection, subset and disjointness. Participants made fewest mistakes using linear diagrams. Wallinger et al. [23] compared the effectiveness of six set tasks with MetroSets, LineSets, and Euler diagrams. On average, MetroSets performed better than the alternatives. Our visualization approach is not directly comparable to any of the presented existing works because UT's embedding is free and dynamically adjustable by the user. This feature stands in contrast to approaches where the embedding is fixed (e.g., KelpFusion [13] for spatial sets) or where it is free but static (e.g., LineSets [1] for network data). Therefore, our primary goal is to derive guidelines for UT instead of comparing UT to existing set visualization approaches.

3 Experiment

This section describes our experiment setup.

3.1 Tasks

We want to test the effectiveness of our visualization technique for cluster and set tasks. Specifically, we formulated the following:

T1 Given a data point as reference, identify other data points in the visualization that are identical to the reference.

T2 Given a set as reference, identify data points in the set.

T3 Given two sets as reference, identify data points in their intersection.

T4 Given a data point as reference, identify other data points that are identical to the reference and also have ≥ 1 set in common.

T1 is meant to be a representative task of visual cluster analysis as it corresponds to *membership identification* in [25]. Multidimensional clusters consist of similar data points. As similarity is a fuzzy concept, we resort to seeking identical data points. This way, we can give clear instructions to participants. **T2** is a basic set task, where participants enumerate members of a set (A1 in [2]). **T3** is a more advanced set task, where participants enumerate an intersection of two sets (A3 in [2]). **T4** is a joint task combining T1 and T3. Each task is answered by a selection of data points.

3.2 Datasets

We generated synthetic datasets following the rules outlined here, each one to be used in one task. A dataset consists of 42 data points, separated into two groups C_d (distractors) and C_t (targets) with $|C_t| = 6$. The reference in tasks T1 and T4 was drawn from C_t. We chose ca. 40 data points to obtain optimally-drawn UT in reasonable time and because glyphs would otherwise be too small. Each dataset has a multidimensional and categorical part. **Multidimensional Part:** Each data point consists of six variables that may take on an integer from 1 to 3. Targets in C_t have the same random assignment of six numbers. Distractors in C_d, on the other hand, have varying random numbers but the Manhattan distance to C_t must be 1–5. This ensures that C_t is not too easy to identify while C_d still has enough variance. We show data points as radar charts (Figure 2). **Categorical Part:** To model categories realistically, we chose metrics from a real-world dataset (specifically the most common genres in the internet movie database): There are six sets in each dataset with predefined frequencies 0.5,0.25,0.15,0.1,0.1,0.1. Each set must contain at least two data points and each data point must be in at least one set. Further, for the T4 reference, there must be at least one set with ≥ 2 data points in C_t and ≥ 1 in C_d. Sets are visualized as lines (Figure 2).

3.3 Stimuli

The stimuli for participants were UT visualizations of the previously described datasets. In particular, the independent variables were the UT *pipeline* setting (2 factors: heuristic or optimal), which controls line routing, and the *weight* α (3 factors: α_0,$\alpha_{0.5}$,α_1), which highlights multidimensional similarity (α_0), set similarity (α_1), or a mixture thereof ($\alpha_{0.5}$) in the glyphs' layout.

We computed UT on an 18×18 grid (plus 1 row/column padding on each side) with DGrid gridification [9] and tree support. We found that projection and network quality metrics are best with the chosen settings. To determine the grid size, we compared projection metrics of square grids with side lengths 12,16,18,20,24 for three random datasets (generated as in Section 3.2 but not used for the experiment). While metrics were slightly better for a side length of 20, we chose 18 as we expected solving stimuli to be more difficult with less whitespace between glyphs. Also, assuming the same viewport size, glyphs could be displayed bigger using a smaller grid. To avoid displaying similar stimuli after another, we randomly mirrored visualizations horizontally, vertically, or both. The line colors were two shades of red, green, and blue, respectively. We selected the colors using Colorgorical [8] to optimize for perceptual distance and name difference.

Participants had to first identify the required task in each question by reading the description. We ensured that each unique description appeared visually distinct to prevent increased mistakes and fatigue. After that, they had to press a button to see the stimulus, i.e., UT. They had to select data points by clicking on glyphs. Once satisfied, another button had to be pressed to finish the time measurement.

3.4 Research Hypotheses

Our hypotheses relate to UT pipelines and its parameters with regard to tasks T1–T4 when considering task accuracy and time. Specifically:

H1 α_0 is more accurate and faster than the alternatives for T1.

H2 α_1 is faster than the alternatives for T2; all have same accuracy.

H3 α_1 is more accurate and faster than the alternatives for T3.

H4 α_0 is more accurate and faster than the alternatives for T4.

H5 $\alpha_{0.5}$ is neither the slowest nor least accurate for any task.

H6 The optimal UT pipeline leads to faster answers in T1–T4.

Our rationale for the hypotheses is the following. We expect α_0 to be best for T1 because it will place glyphs similar to the reference close to it, thus reducing the visual search space compared to $\alpha > 0$ (**H1**). Similar considerations apply to **H2**, but to solve T2, participants only need to follow a single line. Since the support for each set is a tree, a line may branch, but we still expect this task to be rather simple, and differences, therefore, only occur in the answer time. T3, on the other hand, is more complex, and we again expect that α_1 will also lead to better accuracy due to the improved layout (**H3**). To solve T4, a viable strategy for participants is first to find all matching glyphs, then filter those by sets in common with the reference. This strategy is best supported by α_0 (**H4**). $\alpha_{0.5}$ is a compromise between the other two α settings. While we know from prior experiments that set distances may be over-represented at $\alpha_{0.5}$ [16], one could still reasonably expect this layout to be a robust choice for all tasks, i.e., never be the worst alternative (**H5**). Finally, optimally drawn UT visualizations have the same glyph layout as heuristic UT, but shorter lines and fewer crossings. Due to the reduced visual clutter, we expect faster answers using the optimal pipeline (**H6**).

3.5 Participants

We recruited 49 students as part of an undergraduate visualization course. Participation was voluntary, but students had to complete an alternative assignment if they decided against it. Participants (38 male, 11 female) self-reported normal or corrected-to-normal vision. Their median age was 23 within a range of 20–30.

3.6 Procedure

The study was carried out as a within-subject online survey (every participant solved each task T1–T4 with all UT variants) using LimeSurvey. Participants completed it on their own hardware and time. We structured the survey into the following parts:

1. Welcome: The welcome screen acted as a consent form, where we outlined the purpose of our study, contact data, and possible risks when participating.
2. Screening: Participants had to answer questions we used to exclude, e.g., people with dyslexia, uncorrected vision, or too young.
3. Tutorial: We explained individual parts of UT visualizations, the layouts, how the timing works (start and end buttons), and provided examples of T1–T4 they had to solve correctly to advance.
4. Experiment: 24 stimuli and 3 control questions. The control questions were placed at the beginning, middle, and end of the stimuli. Control questions were identical to a T1 stimulus with two differences. First, C_d had a *minimum* Manhattan distance of 6 to C_t (as opposed to maximum distance 5). Second, C_t had a distinctive glyph image. As such, control questions were much easier to answer. We split the stimuli into two groups of 12 so that i) stimuli with same task and α are in separate groups and ii) each group consists of 6 heuristically and 6 optimally drawn stimuli. Stimuli in each group were displayed in randomized order.
5. Post-study Questions: Participants had to rate their preference and confidence in answering T1–T4 using UT with different α. Ratings were on a 5-point Likert scale.
6. Demographics: We collected demographic data to describe our participant sample.

Participants were told to complete the experiment with a compromise of speed and accuracy, i.e., try to answer correctly but also do not spend too much time on each question. They were allowed to take breaks as needed as long as they did so before viewing a stimulus (Section 3.3).

4 Results and Analysis

Figure 1 shows plots of our results. All data and analysis steps can be found in the supplemental material.

4.1 Data

After filtering for incorrect answers to control or screening questions, we were left with the sample described in Section 3.5 and had 49 responses to analyze. We measured the answer accuracy by the F1 score, which combines two quality metrics in information retrieval: Precision (p) and recall (r) [12, Sec. 8.3]. The F1 score is calculated as $2rp/(r+p)$. Time, as outlined in Section 3.3, was measured by the time difference between clicks on the start and end buttons in a question.

4.2 Time and Accuracy

Methods. We analyzed time and accuracy for each task separately. Generally, we checked if the response times are normally distributed visually and with summary statistics, statistical metrics, and the Shapiro-Wilk normality test. If it was not, we Box-Cox-transformed them to a normal distribution. We also checked the homogeneity of variances visually and with Levene's test statistic. No transformation was applied to F1 scores. Instead, we applied nonparametric test statistics. For time measurements, we then ran a two-way analysis of variance (ANOVA) to analyze effects of *weight* (α) and *pipeline* on response time. Where this was the case, we followed with Bonferroni-corrected pairwise t-tests. For accuracy measurements, we used a Kruskal-Wallis rank sum test to analyze the effect of *weight* on accuracy. When this was the case, we ran Dunn's test for multiple comparisons (Bonferroni-corrected).

H1. We can **accept** H1: α_0 is significantly faster and more accurate than the alternatives for task T1. **Time:** The ANOVA revealed a significant effect of *weight* on response time ($F(2) = 126.839, p < 0.001$), but not of *pipeline* or their interaction. The pairwise t-test showed significant differences in the mean response time of α_0 compared to α_1 ($t(194) = -15.34, p < 0.001$) and $\alpha_{0.5}$ ($t(190.5) = -12.58, p < 0.001$). **Accuracy:** The Kruskal-Wallis test showed significant differences between *weight* levels ($\chi^2 = 68.569, df = 2, p < 0.001$). Dunn's test revealed significant differences in the accuracy of α_0 compared to α_1 ($d = -7.47, p < 0.001$) and $\alpha_{0.5}$ ($d = -6.83, p < 0.001$).

H2. We can **reject** H2: We find no differences in accuracy between *weight* levels for task T2. α_0 leads to significantly slower response times, but $\alpha_{0.5}$ is as fast as α_1. **Time:** The ANOVA revealed a significant effect of *weight* on response time ($F(2) = 19.32, p < 0.001$), but not of *pipeline* or their interaction. The pairwise t-test showed significant differences in the mean response time of α_0 compared to α_1 ($t(193.82) = -4.91, p < 0.001$) and $\alpha_{0.5}$ ($t(190.88) = -5.68, p < 0.001$). **Accuracy:** The Kruskal-Wallis test did not reveal differences between *weight* levels ($\chi^2 = 2.0896, df = 2, p = 0.3518$), due to ceiling effects.

H3. We can **reject** H3: α_1 is faster than the alternatives for task T3. $\alpha_{0.5}$ and α_1 are more accurate than α_0. **Time:** The ANOVA revealed a significant effect of *weight* on response time ($F(2) = 59.44, p < 0.001$), but not of *pipeline*. However, the interaction between them is again significant ($F(2) = 14.02, p < 0.001$). Because of the interaction effect, the main components *weight* and *pipeline* cannot be interpreted, and a pairwise t-test is inappropriate. To analyze interaction effects,

we ran an ANOVA and estimated marginal means (EMM) test (post-hoc), for *pipeline* within each level of *weight* and vice versa. The results show significant differences between heuristic and optimal pipelines w.r.t. response time for α_1 ($e(288) = 4.73, p < 0.001$) and $\alpha_{0.5}$ ($e(288) = -2.37, p = 0.018$), but not for α_0. For the response time of *weight* within each *pipeline* level, the analysis revealed significant differences between all *weight* levels for heuristic pipelines. For the optimal pipeline, α_0 differs significantly from the others, but there is no difference between $\alpha_{0.5}$ and α_1. **Accuracy:** The Kruskal-Wallis test showed significant differences between *weight* levels ($\chi^2 = 41.397, df = 2, p < 0.001$). Dunn's test revealed significant differences in the accuracy of α_0 compared to α_1 ($d = 5.58, p < 0.001$) and $\alpha_{0.5}$ ($d = 5.56, p < 0.001$), but not between α_0 and α_1.

H4. We can **reject** H4: There is no difference in time or accuracy for task T4. **Time:** The ANOVA revealed no effect of *weight*, *pipeline*, or their interaction on response time. **Accuracy:** The Kruskal-Wallis test also did not yield significant differences in the accuracy of *weight* levels.

H5. We can **accept** H5 except for T1. For task T1 (tested by H1), $\alpha_{0.5}$ was the slowest and least accurate together with α_1. For task T2 (H2), $\alpha_{0.5}$ was the fastest together with α_1, with no difference in accuracy. For task T3 (H3), $\alpha_{0.5}$ was between the other levels regarding time and more accurate than α_0. For task T4 (H4), there was no difference in accuracy nor time.

H6. We can **reject** H6. The *pipeline* had no significant effect on response time except for $\alpha_{0.5}, \alpha_1$ in task T3. There, the heuristic pipeline performed significantly better with $\alpha_{0.5}$ while the optimal pipeline was significantly better with α_1.

4.3 Preference and Open Feedback

Participants rated their confidence and preference in using a given layout (*weight* factor) for each task on a 5-point Likert scale. We performed a Kruskal-Wallis test to determine differences between *weight* levels. We followed with a Bonferroni-corrected Dunn's test to see the individual comparison results. Confidence ratings poorly correlate (Spearman's rank) with accuracy using heuristic (0.27) or optimal (0.24) UT pipeline. However, a non-negligible correlation (0.66) between preference and confidence scores suggests that participants may not have judged their confidence accurately. Consequently, we only report mean preference ratings. These were generally ordered by which weight can be expected to be most effective for the given task. For T1, the ordering was $\alpha_0 = 5 > \alpha_{0.5} = 3.22 > \alpha_1 = 1.49$ (all significant). For T2, $\alpha_1 = 4.71 > \alpha_{0.5} = 4.06 > \alpha_0 = 2.69$ (all significant). For T3, $\alpha_1 = 4.55 > \alpha_{0.5} = 3.76 > \alpha_0 = 2.04$ (all significant). For T4, $\alpha_1 = 4.55 > \alpha_{0.5} = 3.76 > \alpha_0 = 2.04$ (all significant except $\alpha_{0.5} > \alpha_0$).

Open Feedback. We obtained 24 free text responses. Six mentioned the wording of questions being ambiguous, i.e., that they were in T3 not sure whether to select glyphs based on AND or OR relation of lines. Three mentioned that colors were sometimes hard to distinguish, both across (dark blue and dark red) and within hues (e.g., light and dark red). Some participants reported to find it easier finding shared lines than identical glyphs ("because I can just follow the line"), whereas another said the opposite ("[lines] crossed each other or skipped points, which made it harder for me to follow them"). One participant mentioned the "uncertainty" of $\alpha_{0.5}$ for T1, as it is not clear how much of the reference's neighborhood should be considered.

4.4 Discussion

In **H2**, we explain that $\alpha_{0.5}$ also lead to faster answers by the imbalance of projection quality regarding multidimensional and set distances at $\alpha_{0.5}$. Set distances were better represented at this weight according to our earlier experiments. We suggest the same explanation for accuracy in **H3**, although it is somewhat curious that only either of the two metrics would be affected. Especially interesting regarding **H3** and **H6** is also the interaction between *weight* and *pipeline* in T3, which happens in opposite directions. We check the stimuli for α_1 for clues. In the heuristic pipeline, we can see that the two sought lines are always on the same edge, ordered adjacent to each other, in a connected tree, and in the center of the image. This is not the case for the optimally drawn version of the stimulus, where there is a discontinuity in the support for this intersection, and lines are not adjacent on one edge (see supplemental material). We cannot explain the latter; it may be a bug in our implementation. The situation is reversed for $\alpha_{0.5}$. The optimal pipeline has the two lines on the same edges, ordered next to each other. In the heuristic pipeline, the two sought lines branch a lot and cross other lines. These comparisons show that, depending on the pipeline, there can be substantial visual differences in either direction. However, these mostly did not have an impact on response time. Consequently, there is no disadvantage to using the heuristic pipeline, which can be rendered much more quickly. It was surprising to find no effect in **H4**. A likely explanation we offer is that participants did not follow the strategy we anticipated. We expected them to consider the reference's immediate neighborhood with α_0 and filter these glyphs by shared lines. Instead, they may have done it the other way and followed incident lines while excluding non-matching glyphs. This strategy does not yield benefits with α_0, as irrelevant image parts are scanned, but it works for all layouts. In that case, we would expect more mistakes to happen, which are visible in Figure 1b (though statistically insignificant). As there is no accuracy difference for T2 and little difference in response time, this explanation is consistent with findings for H2 and the preference ratings for T4.

Regarding UT **guidelines**, our cumulative findings [16] indicate that heuristically drawn UT are comparable to optimally drawn counterparts regarding task response time, which makes them attractive for interactive applications. Separate cluster and set tasks can be efficiently and effectively carried out with α_0 (cluster) and α_1 (set) weight settings. For joint analysis, α_0 seems slightly better (T4 in Figure 1a and Figure 1b). DGrid [9] is to be preferred over our implementation of Hagrid [4], and tree supports over paths due to faster heuristics.

5 Limitations

While our tasks aimed to be representative for cluster and set analysis, many more are possible (see, e.g., [2, 25]). We had to restrict their number to keep the experiment's length manageable for participants. The stimuli in our study were the same size, so comparing task performance across more complex stimuli would be interesting and possibly show reduced ceiling effects in T2. We did not control for the visual complexity of stimuli, as they are challenging to measure. Another limitation is the participant sample, as it is not only drawn from the WEIRD (western, educated, industrialized, rich, democratic) population [11] but also skewed towards males and young adults. It is unclear how much our findings would translate to other populations. Finally, comparisons to other set visualization approaches (Section 2.2) would highlight which to choose when.

6 Conclusion

In this paper, we presented a controlled experiment to determine the effect of UT visualization parameters on cluster and set analysis tasks. Thus, we tackle joint visual cluster and set analysis, which has not been explored so far. We could answer our research question: Cluster and set tasks work as expected at the extremes of UT's weight parameter. α_0 supports cluster membership identification [25] while α_1 supports finding members of a set or of an intersection of two sets. The middle setting $\alpha_{0.5}$ was also a robust choice for set-related tasks. Mostly, optimally drawn UT do not lead to faster answers. Contrary to our expectations, neither visualization style was best for the joint task T4, indicating that further research is required on how to enable combined visual cluster and set analysis.

Acknowledgments

We thank our anonymous study participants for their valuable time. This research was supported by EU Horizon 2020 grant 1752849 and Vienna Science and Technology Fund (WWTF) grant [10.47379/ICT19035].

References

[1] B. Alper, N. Riche, G. Ramos, and M. Czerwinski. Design Study of LineSets, a Novel Set Visualization Technique. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2259–2267, Dec. 2011. doi: 10.1109/TVCG.2011.186 2

[2] B. Alsallakh, L. Micallef, W. Aigner, H. Hauser, S. Miksch, and P. Rodgers. The State-of-the-Art of Set Visualization. *Computer Graphics Forum*, 35(1):234–260, Feb. 2016. doi: 10.1111/cgf.12722 2, 4

[3] P. Chapman, G. Stapleton, P. Rodgers, L. Micallef, and A. Blake. Visualizing Sets: An Empirical Comparison of Diagram Types. In T. Dwyer, H. Purchase, and A. Delaney, eds., *Diagrammatic Representation and Inference*, vol. 8578 of *Lecture Notes in Computer Science*, pp. 146–160. Springer, Berlin, Heidelberg, 2014. doi: 10.1007/978-3-662-44043-8_18 2

[4] R. Cutura, C. Morariu, Z. Cheng, Y. Wang, D. Weiskopf, and M. Sedlmair. Hagrid — Gridify Scatterplots with Hilbert and Gosper Curves. In *The 14th International Symposium on Visual Information Communication and Interaction*, pp. 1–8, Article No.: 1. ACM, Potsdam Germany, Sept. 2021. doi: 10.1145/3481549.3481569 4

[5] K. Eckelt, A. Hinterreiter, P. Adelberger, C. Walchshofer, V. Dhanoa, C. Humer, M. Heckmann, C. Steinparz, and M. Streit. Visual Exploration of Relationships and Structure in Low-Dimensional Embeddings. *IEEE Transactions on Visualization and Computer Graphics*, 2022. doi: 10.1109/TVCG.2022.3156760 2

[6] M. Espadoto, R. M. Martins, A. Kerren, N. S. T. Hirata, and A. C. Telea. Towards a Quantitative Survey of Dimension Reduction Techniques. *IEEE Transactions on Visualization and Computer Graphics*, 27(3):2153–2173, 2019. doi: 10.1109/TVCG.2019.2944182 2

[7] R. Etemadpour, R. Motta, J. G. d. S. Paiva, R. Minghim, M. C. F. de Oliveira, and L. Linsen. Perception-Based Evaluation of Projection Methods for Multidimensional Data Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 21(1):81–94, Jan. 2015. doi: 10.1109/TVCG.2014.2330617 2

[8] C. C. Gramazio, D. H. Laidlaw, and K. B. Schloss. Colorgorical: Creating discriminable and preferable color palettes for information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):521–530, Jan. 2017. doi: 10.1109/TVCG.2016.2598918 2

[9] G. M. Hilasaca, W. E. Marcílio-Jr, D. M. Eler, R. M. Martins, and F. V. Paulovich. A Grid-based Method for Removing Overlaps of Dimensionality Reduction Scatterplot Layouts. *IEEE Transactions on Visualization and Computer Graphics*, Oct. 2023. doi: 10.1109/TVCG.2023.3309941 2, 4

[10] J. M. Lewis, L. van der Maaten, and V. R. de Sa. A Behavioral Investigation of Dimensionality Reduction. In *Proceedings of the 34th Annual Conference of the Cognitive Science Society*, pp. 671–676. Cognitive Science Society, 2012. 2

[11] S. Linxen, C. Sturm, F. Brühlmann, V. Cassau, K. Opwis, and K. Reinecke. How WEIRD is CHI? In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–14, Article No.: 143. ACM, Yokohama Japan, May 2021. doi: 10.1145/3411764.3445488 4

[12] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, 2008. 3

[13] W. Meulemans, N. H. Riche, B. Speckmann, B. Alper, and T. Dwyer. KelpFusion: A Hybrid Set Visualization Technique. *IEEE Transactions on Visualization and Computer Graphics*, 19(11):1846–1858, Nov. 2013. doi: 10.1109/TVCG.2013.76 2

[14] C. Morariu, A. Bibal, R. Cutura, B. Frénay, and M. Sedlmair. Predicting User Preferences of Dimensionality Reduction Embedding Quality. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):745–755, Jan. 2023. doi: 10.1109/TVCG.2022.3209449 2

[15] L. G. Nonato and M. Aupetit. Multidimensional Projection for Visual Analytics: Linking Techniques with Distortions, Tasks, and Layout Enrichment. *IEEE Transactions on Visualization and Computer Graphics*, 25(8):2650–2673, Aug. 2019. doi: 10.1109/TVCG.2018.2846735 2

[16] N. Piccolotto, M. Wallinger, S. Miksch, and M. Bögl. UnDRground Tubes: Exploring Spatial Data with Multidimensional Projection and Set Visualization. *IEEE Transactions on Visualization and Computer Graphics*. to appear. Preprint: https://osf.io/zgphx. 1, 3, 4

[17] P. Rodgers, G. Stapleton, B. Alsallakh, L. Michallef, R. Baker, and S. Thompson. A task-based evaluation of combined set and network visualization. *Information Sciences*, 367–368:58–79, Nov. 2016. doi: 10.1016/j.ins.2016.05.045 2

[18] N. Saeed, H. Nam, M. I. U. Haq, and D. B. Muhammad Saqib. A Survey on Multidimensional Scaling. *ACM Computing Surveys*, 51(3):1–25, Article No.: 47, May 2018. doi: 10.1145/3178155 1

[19] M. Sedlmair, T. Munzner, and M. Tory. Empirical Guidance on Scatterplot and Dimension Reduction Technique Choices. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2634–2643, Dec. 2013. doi: 10.1109/TVCG.2013.153 2

[20] M. Sedlmair, A. Tatu, T. Munzner, and M. Tory. A Taxonomy of Visual Cluster Separation Factors. *Computer Graphics Forum*, 31(3pt4):1335–1344, 2012. doi: 10.1111/j.1467-8659.2012.03125.x 2

[21] A. Tatu, P. Bak, E. Bertini, D. Keim, and J. Schneidewind. Visual quality metrics and human perception: An initial study on 2D projections of large multidimensional data. In *Proceedings of the International Conference on Advanced Visual Interfaces*, pp. 49–56. ACM, Roma Italy, May 2010. doi: 10.1145/1842993.1843002 2

[22] E. Wall, M. Agnihotri, L. Matzen, K. Divis, M. Haass, A. Endert, and J. Stasko. A Heuristic Approach to Value-Driven Evaluation of Visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):491–500, Jan. 2019. doi: 10.1109/TVCG.2018.2865146 1

[23] M. Wallinger, B. Jacobsen, S. Kobourov, and M. Nöllenburg. On the Readability of Abstract Set Visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 27(6):2821–2832, June 2021. doi: 10.1109/TVCG.2021.3074615 2

[24] L. Wilkinson, A. Anand, and R. Grossman. High-Dimensional Visual Analytics: Interactive Exploration Guided by Pairwise Views of Point Distributions. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1363–1372, Nov. 2006. doi: 10.1109/TVCG.2006.94 2

[25] J. Xia, Y. Zhang, J. Song, Y. Chen, Y. Wang, and S. Liu. Revisiting Dimensionality Reduction Techniques for Visual Cluster Analysis: An Empirical Study. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):529–539, Jan. 2022. doi: 10.1109/TVCG.2021.3114694 2, 4

[26] D. Xu and Y. Tian. A Comprehensive Survey of Clustering Algorithms. *Annals of Data Science*, 2(2):165–193, June 2015. doi: 10.1007/s40745-015-0040-1 1