Set Visualization

A Survey of Set Visualization Techniques

Una Ibrahimpasic, Eemil Hukkanen, Magne Tenstad, Azra Bajramovic

Information Visualisation SS 2024
Graz University of Technology

13 May 2024

Abstract

Set visualization enables researchers to understand relationships and patterns within set-typed data. This survey paper provides an overview of modern set visualization techniques. The paper begins with an introduction to set-typed data. A taxonomy of tasks is presented, to be used to compare the visualization techniques. The techniques are grouped into five categories: Venn and Euler diagrams, matrix diagrams, node-link diagrams, overlay diagrams, and aggregation diagrams. For each category, the most noteworthy techniques from the last ten years is presented. The discussion then compares their effectiveness in different use-cases, and especially the tasks within the established taxonomy.

© Copyright 2024 by the author(s), except as otherwise noted.

This work is placed under a Creative Commons Attribution 4.0 International (CC BY 4.0) licence.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This survey builds on a previous survey by Alsallakh et al. [2016], bringing the collection of set visualization techniques covered there up to date. Techniques covered in the previous survey will not be covered in detail, apart from in comparision to the new techniques. The images used in this survey paper have been created by the authors of the survey, using available versions of the corresponding software.

1.1 Set-Typed Data

Sets are collections of unique objects, grouped by specific properties [Alsallakh et al. 2016]. An element can belong to one or more sets, but can appear only once inside a single set. A set does not impose any specific order on its elements, i.e. the arrangement of elements within a set is irrelevant. Sets can overlap, which leads to relations between them such as inclusion, exclusion, and intersection. Data involving set memberships of elements is referred to as *set-typed data*. Both the elements and the sets can have additional metadata attributes associated with them, all of which can play a role in data analysis.

Confusion sometimes arise between sets and categories. They both separate data into different groups based on specific properties, and seemingly possess similar characteristics. Indeed, they are often visualised using the same or similar techniques. Based on a helpful email discussion with Alsallakh [2024], the following distinction is drawn. A categorical attribute is single-valued (e.g. `hair-color=" blond"`), taking exactly one value (e.g. `blond`, `brown`, or `red`). Sets of categorical items do not overlap (e.g. blondes, brunettes, redheads). In set-typed data, however, an attribute is multi-valued, it can take a list of values (e.g. `languages-spoken="English,German,French"`). The sets of English speakers and German speakers can intersect. When choosing which representation to use, it is best to align with the goal of the intended analysis.

1.2 Sample Datasets

In order to evaluate the new set visualization techniques, two datasets were chosen. The smaller dataset contains data on 49 European countries and their membership of the following organisations: EAA, EU, EFTA, and Schengen [Netherlands 2024]. The second larger dataset contains data on 1000 movies with 12 different attribute fields [VCG 2024]. The smaller dataset was used to evaluate visualisation techniques which struggle with a large number of sets and set relations without looking cluttered, while the larger set allowed for evaluation of attribute-related tasks for techniques which allow for more in-depth analysis of the datasets.

1.3 Taxonomy of Tasks

Alsallakh et al. [2016] proposed a taxonomy of tasks which defines the possible functionality of a set visualisation. Their taxonomy is split into three categories: *Tasks related to elements*, *Tasks related to sets and set relations*, and *Tasks related to element attributes*. The taxonomy will be used later to compare the set visualization techniques presented in this survey. The tasks is presented below are taken directly from Alsallakh et al. [2016].

1.3.1 Tasks Related to Elements

The following tasks are concerned with the membership of elements in sets:

- **A1**: Find/select elements belonging to a specific set.
- **A2**: Find sets containing a specific element.
- **A3**: Find/select elements based on their set memberships: e.g. elements in A and in B, but not in C.
- **A4**: Find/select elements in a set with a specific set membership degree: e.g. elements exclusive to the set or that also belong to two other sets.
- **A5**: Filter out elements based on their set memberships.
- **A6**: Filter out elements based on their set membership degrees: e.g. filtering out elements exclusive to their sets, to focus on shared elements.
- **A7**: Create a new set that contains certain elements.

1.3.2 Tasks Related to Sets and Set Relationships

The following tasks are concerned with higher level reasoning about the sets, without regard for their elements:

- **B1**: Find out the number of sets in the set family.
- **B2**: Analyse inclusion relations: e.g. find out if a set A is fully included in B, or in $B \cap C$, or in $B \cup C$.
- **B3**: Analyse inclusion hierarchies: e.g. find out if A is included in B, and B in turn is included in C (and so on).
- **B4**: Analyse exclusion relations: e.g. find out if A does not intersect B, or $B \cap C$, or $B \cup C$.
- **B5**: Analyse intersection relations: e.g. find out if a certain pair of sets overlap, or if a certain group of sets overlap (i.e. have a non-empty intersection).
- **B6**: Identify intersections between k sets.
- **B7**: Identify the sets that constitute a certain intersection.
- **B8**: Identify set intersections contained in a specific set.
- **B9**: Identify the set with the largest/smallest number of pairwise set intersections.
- **B10**: Analyse and compare set- and intersection cardinalities: e.g. estimate $|A|$ or $|A \cap B|$, compare $|A|$ with $|B|$, or $|B \cap C|$, or $|B \cup C|$ and identify the set or set intersection with the largest or smallest cardinality.
- **B11**: Analyse and compare set similarities: e.g. find out which pairs of sets exhibit high or low similarity according to some similarity measure.

- **B12**: Analyse and compare set exclusiveness: e.g. find out if A contains more exclusive elements than B, or more elements shared with 1, 2, or 3 other sets.
- **B13**: Highlight specific sets, subsets or set relations: e.g. to emphasize them, and deemphasize the remaining data.
- **B14**: Create a new set using set-theoretic operations: e.g. create the complement of A, or $A \setminus B$ as a new set to compare with other sets.

1.3.3 Tasks Related to Element Attributes

The following tasks are concerned with how element memberships and attributes interrelate:

- **C1**: Find out the attribute values of a certain element.
- **C2**: Find out the distribution of an attribute in a certain set or subset: this aims to understand how the attribute correlates with element membership of this set. Sometimes, the two attributes have a spatial reference and the elements are positioned accordingly as in maps or scatter plots. In this case, the task supports estimating the spatial distribution of a set.
- **C3**: Compare the attribute values between two sets or subsets: e.g. the attribute distributions in two sets can be compared against each other. Alternatively, summary values can be compared such as the mean, the median or the dominant category.
- **C4**: Analyse the set memberships for elements having certain attribute values: e.g. find out if these elements appear more frequently or less often in certain sets/subsets.
- **C5**: Create a new set out of elements that have certain attribute values: this set represents a query on the elements based on their attributes.

Chapter 2

Set Visualization Techniques

In their survey, Alsallakh et al. [2016] essentially propose six categories of set visualization techniques:

- Venn and Euler Diagrams
- Matrix Diagrams
- Node-Link Diagrams
- Overlay Diagrams
- Aggregation Diagrams
- Scatter

These categories group the set visualization techniques according to specific properties. In this survey paper, the first five categories are used, since no techniques were considered to fall into the category Scatter.

2.1 Venn and Euler Diagrams

A Venn Diagram is a type of set visualization that uses closed curves, such as circles or ellipses, to represent sets and their relationships. Venn diagrams must show all possible combinations of curve overlaps, making them visually complex as more sets are depicted [Alsallakh et al. 2016], as shown in Figure 2.1.

A Euler Diagram is similar to a Venn Diagram in that it uses closed curves to represent sets and their relationships. Unlike Venn diagrams, Euler diagrams have no restrictions on how the curves overlap, allowing them to represent set inclusion, exclusion, and intersections [Alsallakh et al. 2016], as shown in Figure 2.1.

InteractiVenn is a groundbreaking technique that offers flexibility in interacting with Venn diagrams, accommodating upto six sets [Heberle et al. 2015]. It features a user-friendly interface for constructing Venn diagrams and enables the analysis of set unions while maintaining the diagram's structure, as shown in Figure 2.3. This capability is particularly valuable for identifying both differences and similarities among sets. Users can explore set unions using either a tree or a list. Furthermore, the tool allows users to access subset elements, save and load sets for further analysis, and export diagrams in vector and image formats. Alsallakh et al. [2016] highlighted that Venn diagrams often become overly complex. However, InteractiVenn addresses this issue by providing a solution that allows for the exploration of complex relationships in a more intuitive and manageable way. For example,q by offering possibility to create unions.

EulerMerge is a technique for presenting Euler diagrams. It operates through an iterative process,

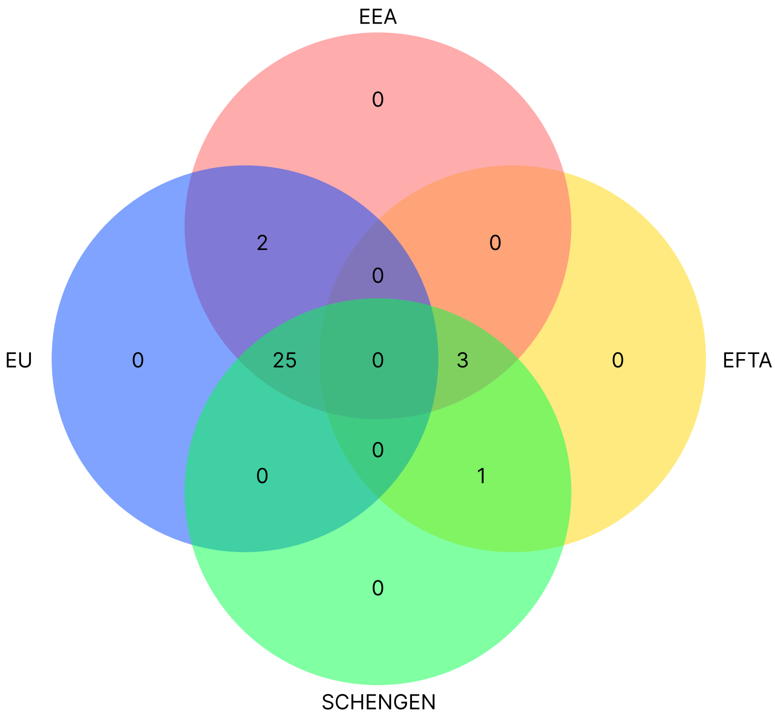

Figure 2.1: Venn diagram with countries dataset. [Image created by Magne Tenstad.]

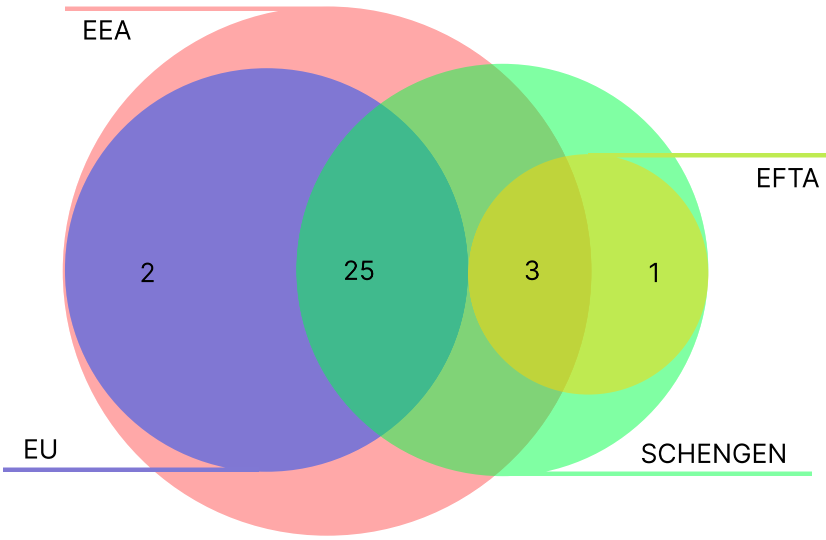

Figure 2.2: Euler diagram with countries dataset. [Image created by Magne Tenstad.]

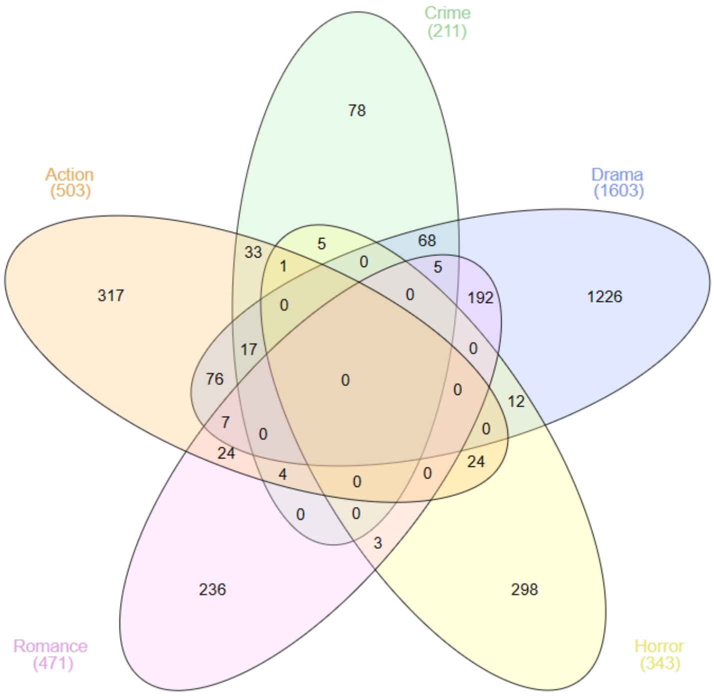

Figure 2.3: InteractiVenn with a movie dataset. [Screenshot of InteractiVenn [Heberle et al. 2024], made by the authors of this survey.]

selecting pairs of sets and replacing them with their union. This process is aimed at reducing the complexity of Euler diagrams while preserving their structure. The algorithm maintains a set of zones, which partition the universe based on the elements they contain, forming the foundation for the simplification process [Yan et al. 2024].

2.2 Matrix Diagrams

Several techniques use matrices to visualize set memberships. This is usually an attempt at improving readability by having a clearly defined structure, as opposed to the free-flowing topology of some Euler and Venn diagrams. Since matrices are limited when it comes to showing attribute data, these visualizations often include supplementary panels with aggregated information about the elements selected in the matrix.

Alsallakh et al. [2016] classifies all techniques that include a grid to be matrix-based. However, not all these grids are true rectangles. Additionally, their rows and columns do not necessarily provide any information. One example of this is OnSet [Sadana et al. 2014]. In this survey, such techniques will be categorized as simply *grid-based* and will not be further discussed. True *matrix-based* techniques assign meaning to their rows and columns.

UpSet visualizes intersections and their aggregates, the number of elements, and attribute statistics [Lex et al. 2014]. While the visualization as a whole includes different sub-panels with histograms, bar charts, and scatter plots, the main innovation is the *combination matrix*. A set is assigned to each column of the matrix, and each row represents an intersection between a number of these sets. UpSet has multiple implementations [Lex 2024]. The most easily accessible is UpSet 2, a re-implementation of UpSet using modern web technologies. According to [Gadhave et al. 2019], UpSet 2 improves on UpSet by improving in three areas: easing adoption, easing sharing, and enable integration. Figure 2.4 shows a screenshot of UpSet 2, with the movie dataset.

Aggreset was developed to improve on scalability and to support rich, contextual exploration of set-typed data [Yalçin et al. 2016]. Figure 2.4 shows a screenshot of AggreSet with the movie dataset.

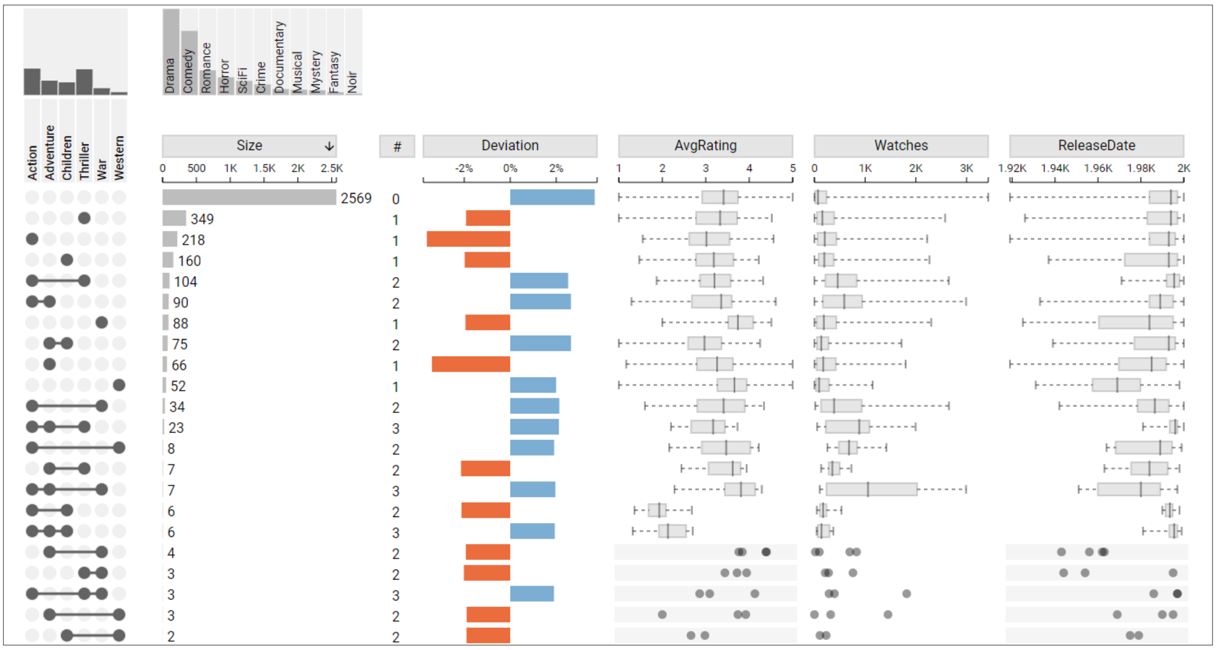

Figure 2.4: UpSet 2 with the movie dataset. [Screenshot of UpSet 2 [VDL 2024], made by the authors of this survey.]

Instead of assigning sets only to columns, as UpSet does, AggreSet assigns sets to both columns and rows. Consequently, each cell in the matrix represents an intersection between two sets, given by its row and column. In order to analyze intersections between more than one set, the user has to select the sets of interest. This is less flexible than the UpSet combination matrix which displays all possible intersections without the need for interaction. On the other hand, the size of the AggreSet matrix does not grow as tall as the UpSet one. AggreSet also includes additional histograms, which display aggregated attributes for the whole dataset as well as the selection.

2.3 Node-Link Diagrams

Node-link diagrams define a category of set visualisation techniques which model the membership relations between elements and sets as edges of a bipartite graph whose nodes represent elements or sets [Alsallakh et al. 2016]. These techniques often employ links of varying thickness in order to show the similarity between the sets. Due to their design, these techniques usually do not fulfill tasks in the third category of the taxonomy defined in the beginning of the survey, which are element attribute related tasks. Three new techniques are introduced here in the node-link category: BranchingSets [F. Paduano et al. 2016], MetroSets [Jacobsen et al. 2021], and NetSet [Park et al. 2017].

BranchingSets was introduced as a solution for problems representing large datasets of biological pathway networks [F. Paduano et al. 2016]. It is an interactive visualization technique which eases the process of pattern recognition and relationship identification. The BranchingSets visualization enables identification of nodes and links in different categories, inspection of node properties, and finding connections between nodes based on user-defined criteria. The user can move nodes around and toggle specific graphs or collapse sub graphs in order to simplify the visualization of complex networks. Figure 2.6 shows the Pathway Explorer [A. Paduano 2024] with a biological pathways dataset. Since the demo website for BranchingSets does not support the import of an original dataset, the evaluation of this visualization technique is restricted to the presupplied dataset.

MetroSets was designed to use the metaphor of metro (underground) maps [Jacobsen et al. 2021]. This is an interactive visualization technique, in which the sets are represented by metro lines and the

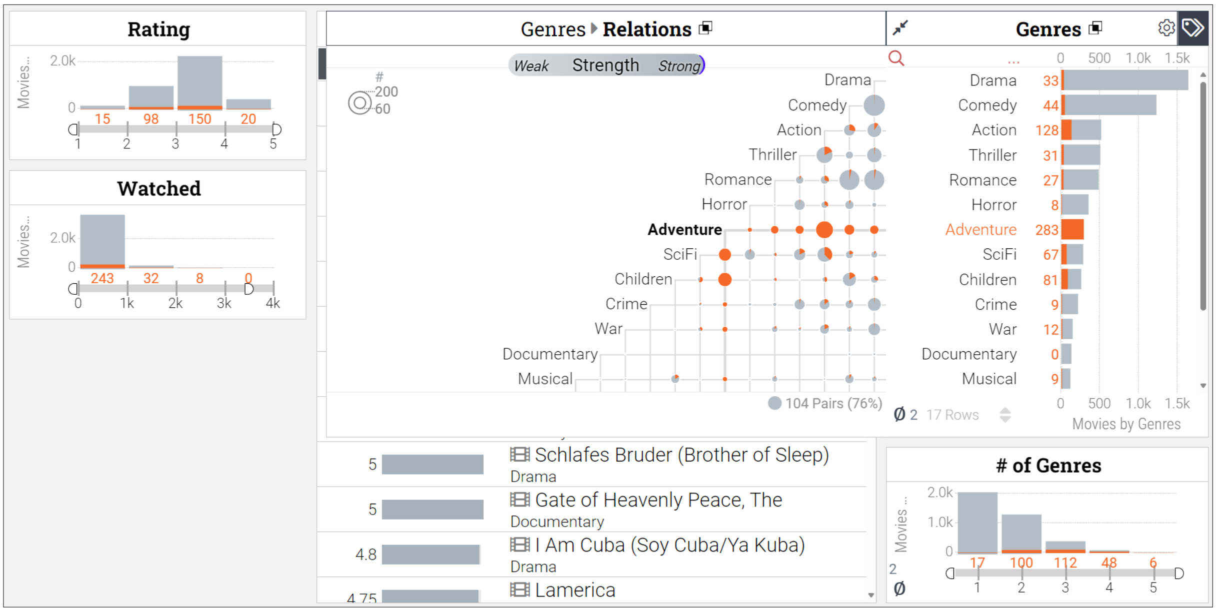

Figure 2.5: AggreSet with movie dataset. [Screenshot of AggreSet [Yalçın et al. 2024], made by the authors of this survey.]

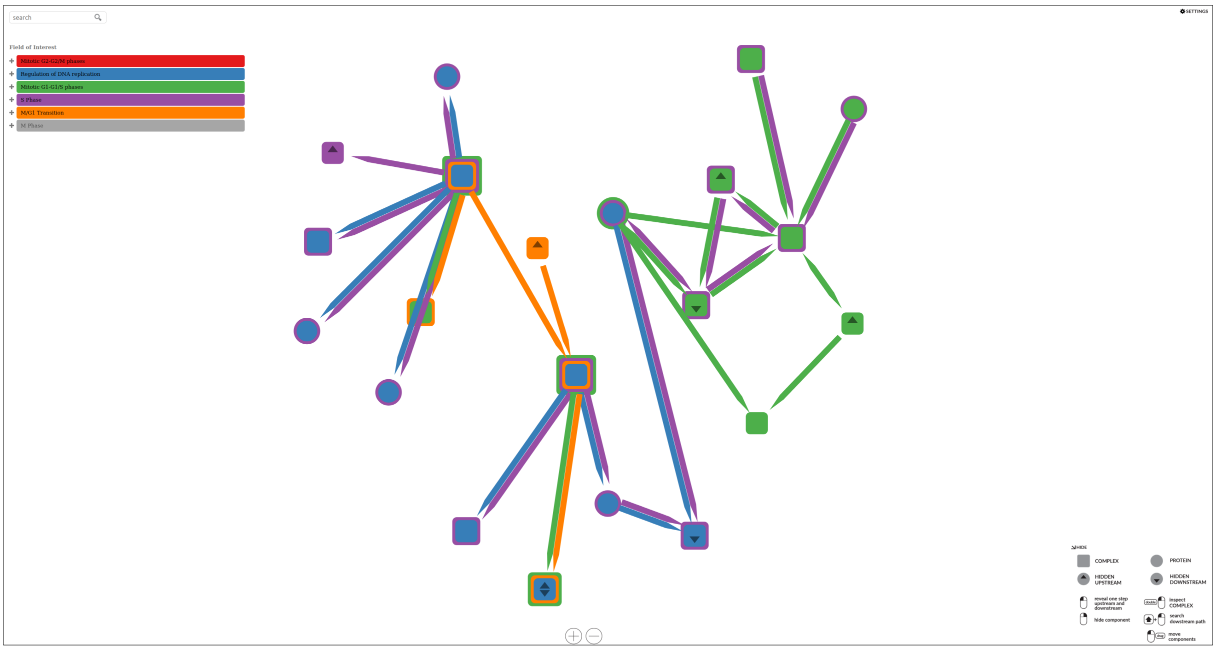

Figure 2.6: BranchingSets demo with biological pathways dataset. [Screenshot of BranchingSets [A. Paduano 2024], made by the authors of this survey.]

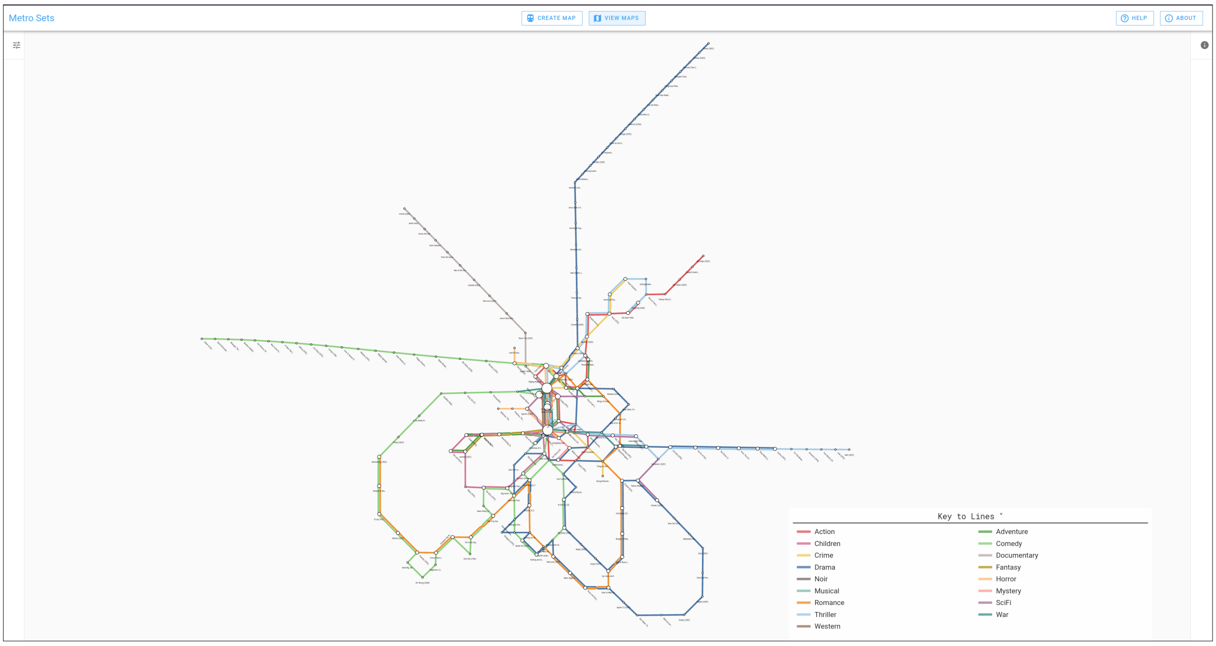

Figure 2.7: MetroSets demo using the movie dataset. [Screenshot of MetroSets [VUT 2024], made by the authors of this survey.]

elements as metro stations along the lines. Sets are distinguishable by the different color of the line, and elements which belong to more than one set correspond to interchange stations. Lines or vertices can be emphasized by hovering over them, whereby lower opacity is applied to all other elements to deemphasize them. Tooltips in the visualization display additional information such as vertex label or other data attributes. Finally, filtering options include intersection, union, complement, symmetric difference, and subtract mode. Figure 2.7 is a visualisation of the movie dataset using MetroSets [VUT 2024]. For the movies dataset to be visualized by MetroSets, attributes of movies which were categorical in nature had to be removed only the information about the genres to which a movie belongs remained. Additionally, the dataset was manipulated to accommodate the rstriction of MetroSets that each set (genre) has to have at least two elements.

NetSet was developed with the aim to tackle the limitations of both matrix-based and node-link visualizations [Park et al. 2017]. NetSet employs both a bipartite network and the matrix-like interface of UpSet. NetSet facilitates the analysis of set intersections by allowing users to select desired sets on a network, displaying all intersections in a matrix, as done in UpSet. It employs bidirectional linking and brushing to coordinate the network and matrix views, enabling users to understand the relative positions of sets while examining intersection details. Additionally, NetSet offers different set views based on attributes or centrality, providing insights into the structure of the set system and allowing manipulation of set operations through a query interface.

2.4 Overlay Diagrams

As described by Alsallakh et al. [2016], data often needs to be analyzed in context of other data features, making set membership only a secondary concern. For instance, spatially referenced elements are often seen in relation to a map that offers additional location-based information. Similarly, nodes in a graph or points in a scatter plot can be contextualized by adding labels, colors, lines, and shapes. Alsallakh et al. [2016] identify three strategies to overlay set membership information onto existing visualizations like maps, scatter plots, and graphs:

- *Regions*: Enclosing elements to define distinct regions. Overlapping regions show multiple set memberships.

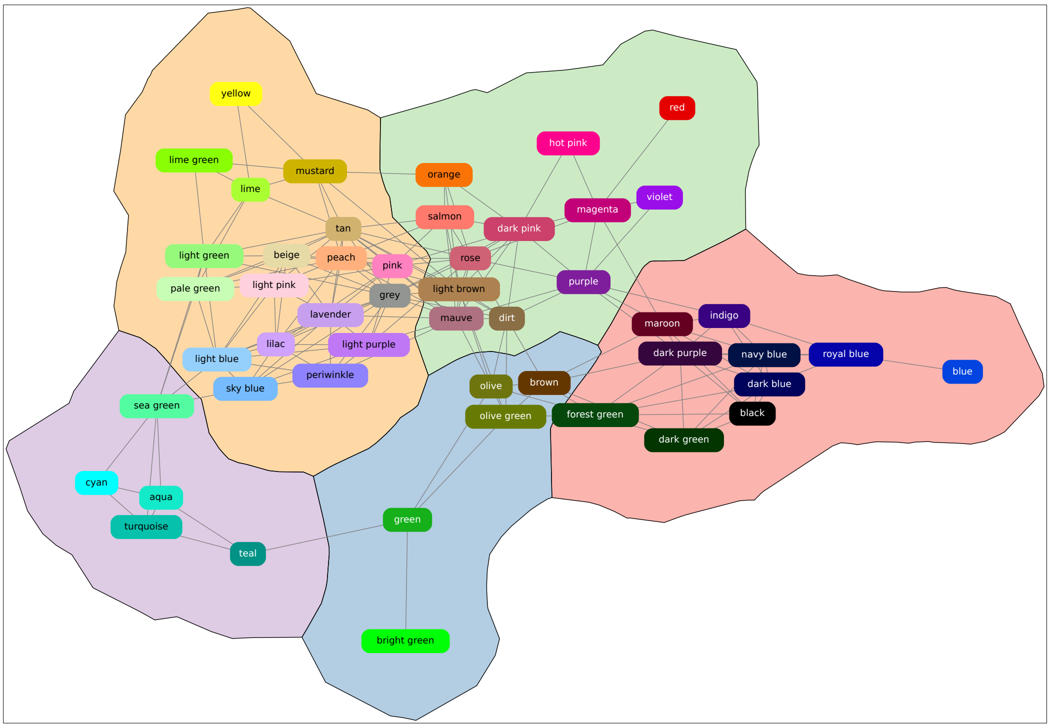

Figure 2.8: MapSets with the pre-supplied colors dataset, showing five sets of colors. [Screenshot of MapSets [Kobourov et al. 2024]. Used with kind permission of Keith Andrews.]

- *Lines*: Using lines to represent set membership.
- *Glyphs*: Placing icons, shapes, and/or colors next to data points to show set membership.

The three can be combined into more complex visualizations. Four new overlay techniques are intoduced here: F2-BubbleSet [Wang et al. 2022], MapSets [Efrat et al. 2014], GridSets [Chung et al. 2022], and TimeSets [Nguyen et al. 2016].

F2-BubbleSets [Wang et al. 2022] was introduced as an interactive counterpart to BubbleSets [Collins et al. 2009]. The technique features simultaneous construction of spanning trees with relation-aware energy fields whose contour widths are based on set proximity. It also blends line and region visualizations by having overlapping sets enclosed in regions. Interactions include node and edge manipulation, control point adjustment, and direct contour control for user customization.

MapSets technique is used to visualize embedded and clustered graphs [Efrat et al. 2014], where the vertices have fixed positions (e.g. geographic locations) and are divided into clusters. Clusters are represented by contiguous, non-overlapping, convex regions, optimized to use the shortest curve to connect all cluster points within the regions (the "minimum ink" principle). The authors provide an online implementation of MapSets, along with three other similar techniques (GMap, BubbleSets, and KelpFusion) [Kobourov et al. 2024]. Figure 2.8 shows a MapSets visualisation of its pre-supplied colors dataset.

GridSet aims to address the difficulties in analyzing individual elements and attributes while considering set memberships, manage visual complexity and facilitate set operations [Chung et al. 2022]. Each set is displayed as a grid containing small glyphs that represent the elements. Elements within set grids are organized using a treemap layout, which arranges them based on their intersections with other sets. Elements can be duplicated across multiple set grids to show their membership in different sets. Multiple element attributes can be encoded using multiple glyphs. GridSet offers a plethora of interactive features, including dynamic set addition/removal and spatial organization via drag-and-drop, combined set operations directly on the set grids, dynamic queries to filter elements and intersections based on attribute values or intersection degrees, and the creation of user-defined sets based on selected elements or attributes.

TimeSets was designed to analyze complex temporal datasets by showing set relationships among individual events [Nguyen et al. 2016]. It groups events that share a topic, in temporal order. The level of detail per event is based on the available display space. Events are represented by a circle next to the event label. Events belonging to the same set are placed close together vertically. Each set has a unique colored background connecting its events. Sets are stacked vertically, with a maximum of three horizontal layers per set. Events shared between two neighboring sets are placed in a shared layer with a color gradient background transitioning between the two set colors. Events shared between non-neighboring sets are duplicated in both sets. Interactions include details on hover, time range zooming along the timeline, set filtering and set visibility toggle.

2.5 Aggregation Diagrams

Aggregation diagrams hide individual elements. They employ frequency representations to show the number of elements in different sets and subsets [Alsallakh et al. 2016]. These frequency representations are usually area-proportional, such that different subsets are intuitively comparable. Since individual elements are not considered, it is not uncommon to discard element attributes. However, they may also be aggregated and displayed as a sum or mean.

PowerSet aims to provide a compact and scalable overview of all set intersections in a set system [Alsallakh and Ren 2017]. In a Treemaps-based fashion, intersections are shown as area-proportional rectangles. They are sorted by size on the x-axis, and number of intersections on the y-axis. Additionally, they may be colored by (the mean of) an attribute of choice. This results in a clear overview of the most common intersections, but is more difficult to read for the less common ones.

Radial Sets is a tool for visual analysis of large overlapping sets [Alsallakh et al. 2013]. Figure 2.9 shows it with the movie dataset. The sets are distributed as segments along the circle circumference. The width of each segment corresponds to the set size. Additionally, lines are drawn between sets that overlap, with thickness according to the number of elements in their intersection. Inner subsegments represent intersections of increasing degrees from outside in. Radial Sets works better than PowerSet for analyzing *specific* set intersections, while PowerSet provides a better ordered overview of *all* intersections.

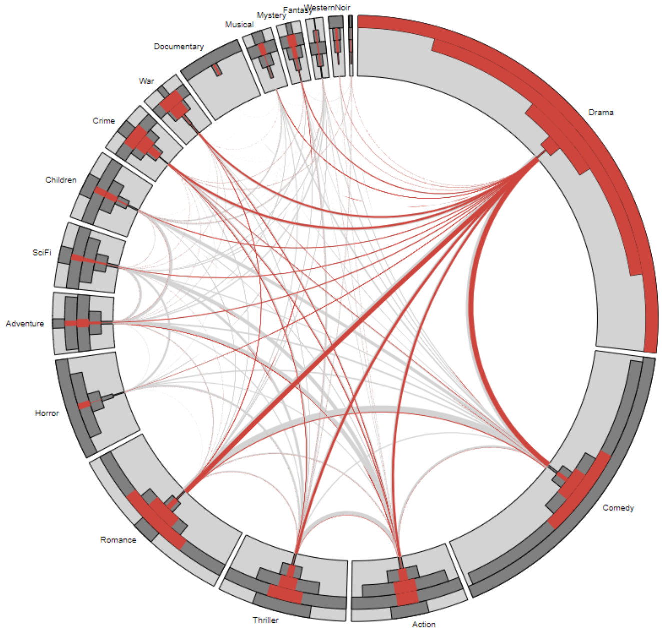

Figure 2.9: Radial Sets with the movie dataset. [Screenshot of Radial Sets demo taken by the authors of this paper. Software kindly provided by Bilal Alsallakh.]

Chapter 3

Summary

Each set visualization technique possesses strengths and weaknesses that should be taken into consideration when choosing the appropriate data representation. From Alsallakh et al. [2016], Table 3.1 provides an overview of the strengths and weaknesses of the five groups of techniques.

For the thirteen new set visualization techniques covered in this survey, Table 3.2 shows which tasks from the taxonomy are supported by each technique. An *x* denotes that the task is supported either by the static visualization itself or through user interaction with the system.

	Strengths	**Weaknesses**
1. Venn and Euler Diagrams	Intuitive when well-matched (little training is required). Represent all standard set relations compactly.	Limited to small number of sets due to clutter and drawability issues. Desired properties not always possible (e.g. convexity).
2. Matrix Diagrams	Fairly scalable both in the number of elements and sets. Do not suffer from edge crossings or topological constraints.	Limited in the set relations they can represent. Revealed membership patterns are sensitive to ordering.
3. Node-Link Diagrams	Visually emphasize elements as individual objects. Show clusters of elements having similar set memberships.	No representation of set relations in element-set diagrams.
4. Overlay Diagrams	Emphasize element and set distributions according to other data features (e.g. map locations).	Often limited in number of elements and sets.
5. Aggregation Diagrams	Highly scalable in number of elements. Some techniques can show how attributes correlate with set membership.	Do not usually show individual elements. Limited in the set relations they can represent.

Table 3.1: Comparison of set visualisation techniques.

Techniques	**Element Related**							**Set Related**														**Element Attribute Related**				
	A1	A2	A3	A4	A5	A6	A7	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12	B13	B14	C1	C2	C3	C4	C5
InteractiVenn	x	x	x						x	x	x	x		x	x		x			n/a		x				
Eulermerge	x	x	x						x	x	x	x		x	x		x			n/a						
UpSet2	x		x	x	x			x				x	x	x	x	x	x	x		x		x	x	x	x	
AggreSet	x		x		x			x	x			x	x	x			x			x		x	x	x		
BranchingSets	x	x	x					x	x	x	x											x	x	x	x	x
MetroSets	x	x	x		x			x	x	x	x	x		x			x			x	x	x		x		
NetSet	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
F2-BubbleSets	x		x		x				x	x	x	x		x			x									
MapSets								x	x	x	x		x	x		x	x	x				x				
GridSets	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
TimeSets	x	x	x		x				x	x	x	x	x		x			x								
PowerSet								x				x	x	x		x	x	x						x		
Radial Sets	x		x	x	x	x		x			x	x	x	x	x	x	x	x	x	x	x		x	x	x	x

Table 3.2: Support for various tasks.

Chapter 4

Concluding Remarks

Sets and set relations are essential in many data analysis scenarios. Numerous techniques have been devised to visualize sets and data related to them in the past decade. This paper serves as a complement to the original survey paper and browser by Alsallakh et al. [2016], updating it with thirteen more recent visualization techniques. These techniques are categorized based on the main visual representation they use to depict set relations.

Additionally, the source code and the content of the original visual browser used to showcase the surveyed techniques has been updated. The source code is is available on GitHub [Tenstad et al. 2024a]. A public live demo is available too [Tenstad et al. 2024b], but does not contain any thumbnail images due to lack of copyright permissions.

Bibliography

Alsallakh, Bilal [2024]. *set vis vs category vis*. Email correspondance. 23 Apr 2024 (cited on page 1).

Alsallakh, Bilal, Wolfgang Aigner, Silvia Miksch, and Helwig Hauser [2013]. *Radial Sets: Interactive Visual Analysis of Large Overlapping Sets*. IEEE Transactions on Visualization and Computer Graphics 19.12 (Dec 2013), pages 2496–2505. ISSN 1077-2626. doi:10.1109/TVCG.2013.184 (cited on page 12).

Alsallakh, Bilal, Luana Micallef, Wolfgang Aigner, Helwig Hauser, Silvia Miksch, and Peter Rodgers [2016]. *The State-of-the-Art of Set Visualization*. Computer Graphics Forum 35.1 (Feb 2016), pages 234–260. ISSN 0167-7055. doi:10.1111/cgf.12722 (cited on pages 1–2, 5, 7–8, 10, 12, 15, 17).

Alsallakh, Bilal and Liu Ren [2017]. *PowerSet: A Comprehensive Visualization of Set Intersections*. IEEE Transactions on Visualization and Computer Graphics 23.1 (Jan 2017), pages 361–370. ISSN 1077-2626. doi:10.1109/TVCG.2016.2598496 (cited on page 12).

Chung, Haeyong, Santhosh Nandhakumar, and Seungwon Yang [2022]. *GridSet: Visualizing Individual Elements and Attributes for Analysis of Set-Typed Data*. IEEE Transactions on Visualization and Computer Graphics 28.8 (Aug 2022), pages 2983–2998. ISSN 1077-2626. doi:10.1109/TVCG.2020.3047111 (cited on page 11).

Collins, Christopher, Gerald Penn, and Sheelagh Carpendale [2009]. *Bubble Sets: Revealing Set Relations with Isocontours over Existing Visualizations*. IEEE Transactions on Visualization and Computer Graphics 15.6 (Nov 2009), pages 1009–1016. ISSN 1077-2626. doi:10.1109/TVCG.2009.122 (cited on page 11).

Efrat, Alon, Yifan Hu, Stephen G. Kobourov, and Sergey Pupyrev [2014]. *MapSets: Visualizing Embedded and Clustered Graphs*. Proc. 22nd International Symposium on Graph Drawing (GD 2014) (Würzburg, Germany). Springer, 24 Sep 2014, pages 452–463. doi:10.1007/978-3-662-45803-7_38 (cited on page 11).

Gadhave, Kiran, Hendrik Strobelt, Nils Gehlenborg, and Alexander Lex [2019]. *UpSet 2: From Prototype to Tool*. Proc. IEEE Information Visualization Conference - Posters (InfoVis 2019) (Vancouver, Canada). 20 Oct 2019. `https://sci.utah.edu/~vdl/papers/2019_infovis_upset.pdf` (cited on page 7).

Heberle, Henry, Gabriela Vaz Meirelles, Felipe R. da Silva, Guilherme P. Telles, and Rosane Minghim [2015]. *InteractiVenn: A Web-Based Tool for the Analysis of Sets through Venn Diagrams*. BMC Bioinformatics 16.1 (22 May 2015). ISSN 1471-2105. doi:10.1186/s12859-015-0611-3 (cited on page 5).

Heberle, Henry, Gabriela Vaz Meirelles, Felipe R. da Silva, Guilherme P. Telles, and Rosane Minghim [2024]. *InteractiVenn - Interactive Venn Diagrams*. 13 May 2024. `https://interactivenn.net/` (cited on page 7).

Jacobsen, Ben, Markus Wallinger, Stephen Kobourov, and Martin Nöllenburg [2021]. *MetroSets: Visualizing Sets as Metro Maps*. IEEE Transactions on Visualization and Computer Graphics 27.2 (Feb 2021), pages 1257–1267. ISSN 1077-2626. doi:10.1109/TVCG.2020.3030475 (cited on page 8).

Kobourov, Stephen, Sergey Pupyrev, and Paolo Simonetto [2024]. *GMAP*. 12 May 2024. `http://gmap.cs.arizona.edu/` (cited on page 11).

Lex, Alexander [2024]. *Implementations of UpSet*. 01 Apr 2024. https://upset.app/implementations/ (cited on page 7).

Lex, Alexander, Nils Gehlenborg, Hendrik Strobelt, Romain Vuillemot, and Hanspeter Pfister [2014]. *UpSet: Visualization of Intersecting Sets*. IEEE Transactions on Visualization and Computer Graphics 20.12 (Nov 2014), pages 1983–1992. doi:10.1109/TVCG.2014.2346248 (cited on page 7).

Netherlands [2024]. *What countries are in the EU, EEA, EFTA and the Schengen area?* Netherlands Worldwide, 13 May 2024. https://netherlandsworldwide.nl/eu-eea-efta-schengen-countries (cited on page 1).

Nguyen, Phong H., Kai Xu, Rick Walker, and BL William Wong [2016]. *TimeSets: Timeline Visualization with Set Relations*. Information Visualization 15.3 (Jul 2016), pages 253–269. doi:10.1177/1473871615605347 (cited on pages 11–12).

Paduano, Angelo [2024]. *Pathways*. 12 May 2024. https://paduano.github.io/pathways/ (cited on pages 8–9).

Paduano, Francesco, Ronak Etemadpour, and Angus G. Forbes [2016]. *BranchingSets: Interactively Visualizing Categories on Node-Link Diagrams*. Proc. 9th International Symposium on Visual Information Communication and Interaction (VINCI 2016) (Dallas, Texas, USA). ACM, 24 Sep 2016, pages 9–16. doi:10.1145/2968220.2968229 (cited on page 8).

Park, Heungseok, Hongjun Lim, Wonjae Lee, and Kyungwon Lee [2017]. *NetSet: A Systematic Integration of Visualization for Analyzing Set Intersections with Network*. Proc. 2017 IEEE Pacific Visualization Symposium (PacificVis 2017) (Seoul, Korea). 18 Apr 2017, pages 26–30. doi:10.1109/PACIFICVIS.2017.8031575 (cited on pages 8, 10).

Sadana, Ramik, Timothy Major, Alistair Dove, and John Stasko [2014]. *OnSet: A Visualization Technique for Large-scale Binary Set Data*. IEEE Transactions on Visualization and Computer Graphics 20.12 (Dec 2014), pages 1993–2002. ISSN 1077-2626. doi:10.1109/TVCG.2014.2346249 (cited on page 7).

Tenstad, Magne, Una Ibrahimpasic, Azra Bajramovic, and Eemil Hukkanen [2024a]. *LitExplorer: A Visual Browser for Literature Collections*. 13 May 2024. https://github.com/info-vis-24/lit-explorer (cited on page 17).

Tenstad, Magne, Una Ibrahimpasic, Azra Bajramovic, and Eemil Hukkanen [2024b]. *Survey Browser*. 13 May 2024. https://info-vis-24.github.io/survey-browser/ (cited on page 17).

VCG [2024]. *UpSet: Movies Dataset*. Visual Computing Group, Harvard University, 12 May 2024. https://github.com/VCG/upset/tree/master/data/movies (cited on page 1).

VDL [2024]. *UpSet - Visualizing Intersecting Sets*. Visualization Design Lab, University of Utah, 13 May 2024. https://upset.multinet.app/ (cited on page 8).

VUT [2024]. *MetroSets*. Vienna University of Technology, 12 May 2024. https://metrosets.ac.tuwien.ac.at/ (cited on page 10).

Wang, Yunhai, Da Cheng, Zhirui Wang, Jian Zhang, Liang Zhou, Gaoqi He, and Oliver Deussen [2022]. *F2-Bubbles: Faithful Bubble Set Construction and Flexible Editing*. IEEE Transactions on Visualization and Computer Graphics 28.1 (Jan 2022), pages 422–432. ISSN 1077-2626. doi:10.1109/TVCG.2021.3114761 (cited on page 11).

Yalçin, M. Adil, Niklas Elmqvist, and Benjamin B. Bederson [2016]. *AggreSet: Rich and Scalable Set Exploration using Visualizations of Element Aggregations*. IEEE Transactions on Visualization and Computer Graphics 22.1 (Jan 2016), pages 688–697. ISSN 1077-2626. doi:10.1109/TVCG.2015.2467051 (cited on page 7).

Yalçın, M. Adil, Niklas Elmqvist, and Ben Bederson [2024]. *AggreSet Visualization Technique*. 13 May 2024. https://gallery.keshif.me/AggreSet/ (cited on page 9).

Yan, Xinyuan, Peter Rodgers, Peter Rottmann, Daniel Archambault, Jan-Henrik Haunert, and Bei Wang [2024]. *EulerMerge: Simplifying Euler Diagrams Through Set Merges*. Diagrams 2024 (Münster, Germany). 2024. doi:10.48550/arXiv.2311.15396 (cited on page 7).