Eurographics Conference on Visualization (EuroVis) (2014)
R. Borgo, R. Maciejewski, and I. Viola (Editors)

STAR – State of The Art Report

Visualizing Sets and Set-typed Data: State-of-the-Art and Future Challenges

Bilal Alsallakh[1], Luana Micallef[2,3], Wolfgang Aigner[1,4], Helwig Hauser[5], Silvia Miksch[1], and Peter Rodgers[3]

[1]Vienna University of Technology, Austria [2]Helsinki Institute for Information Technology HIIT, Finland
[3]University of Kent, United Kingdom [4]St. Pölten University of Applied Sciences, Austria [5]University of Bergen, Norway

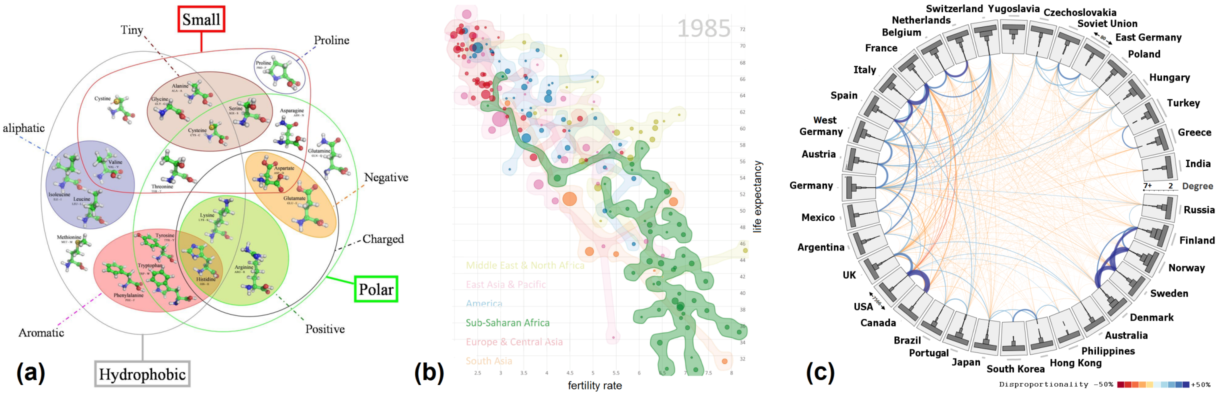


Figure 1: Different set visualizations: (a) An Euler diagram [Pod08], (b) Bubble Sets [CPC09], (c) Radial Sets [AAMH13].

Abstract

A variety of data analysis problems can be modelled by defining multiple sets over a collection of elements and analyzing the relations between these sets. Despite their simple concept, visualizing sets is a non-trivial problem due to the large number of possible relations between them. We provide a systematic overview of state-of-the-art techniques for visualizing different kinds of set relations. We classify these techniques into 7 main categories according to the visual representations they use and the tasks they support. We compare the categories to provide guidance for choosing an appropriate technique for a given problem. Finally, we identify challenges in this area that need further research and propose possible directions to address these challenges.

Categories and Subject Descriptors (according to ACM CCS): H.5.2 [Information Interfaces and Presentation]: User Interfaces—Graphical user interfaces F.4.1 [Theory of Computation]: Mathematical Logic—Set theory

1. Introduction

Data items are often grouped into sets based on specific properties. For instance, Fig. 1a shows amino acids grouped according to known features, while Fig. 1b groups countries by continent. Several relations between the sets are possible, such as: containment, exclusion, and intersection. Analyzing these relations is key to gain information about the behaviour of the entities they represent. Such information might involve which sets have strong overlaps (Fig. 1c), and whether certain data features are responsible for this. A variety of real-world concepts can be modelled using sets, including: club memberships, product features, and employee skill sets. Example questions about such data are: whether certain clubs are exclusive to each other, whether a certain product feature is always in combination with another one, and whether specific skill combinations are highly paid.

Information visualization (InfoVis) offers many opportunities for analyzing sets and their relations. A key challenge in visualizing sets is the potentially large number of possible relations between them, as it grows exponentially with the number of sets. This imposes a severe limit on common representations based on Venn and Euler diagrams. Several InfoVis techniques were proposed to visualize sets using alternative representations. These techniques vary in their scalability limits and in the set-related tasks they support.

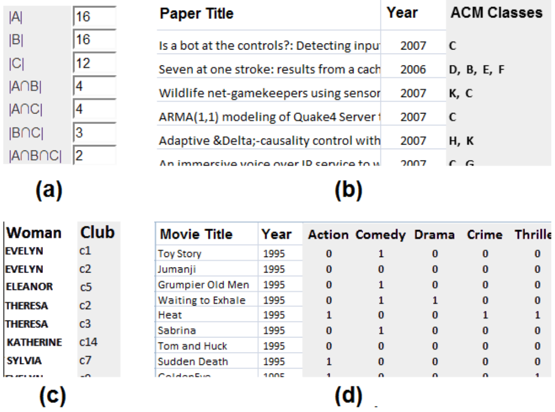
In this report, we survey state-of-the-art techniques for visualizing sets. After discussing several characteristics of set-typed data (Sect. 2) and tasks related to them (Sect. 3), we provide an overview and a categorization of these techniques (Sect. 4) based on the visual metaphors they use. In addition, we compare these techniques according by their advantages and limitations, and by the tasks they support (Sect. 5). Finally, we identify challenges that require future research, along with possible opportunities to tackle them (Sect. 6).

2. Sets and Set-typed Data

Sets have been traditionally studied by mathematicians and logicians as a foundational concept. A set is defined as a collection of unique objects, called the set elements. A key characteristic of this collection is that it does not impose an ordering of the elements. A family of sets, also called a set system, is a collection of subsets of a given set of elements. Such sets potentially overlap, making several relations between the sets possible such as containment, exclusion, and intersection. Cantor formalized *set theory* [Can95] in the 19th century. This theory is concerned with various concepts related to sets, such as set algebra and set operations.

In data analysis, sets have been mainly treated as a collection of data points, such as a subset of rows in a data table. Such subsets are usually used to define training and validation sets, or to store the results of search and clustering algorithms. In addition, set-theoretic operations such as intersection, union, difference, complement, Cartesian product and the power set are extensively used in relational databases to query elements and join multiple data tables.

Despite the ubiquitous usage of sets in data analysis, sets have not been commonly treated as an own data type in InfoVis literature, unlike graphs and hierarchies. Set memberships are rather often abstracted as separate Boolean attributes. Freiler et al. [FMH08] pointed to this lack, and proposed several ideas to support set-enabled visualizations. Treating set families as an elementary data type would contribute to a better understanding of their characteristics and the challenges associated with visualizing them. We refer to data that involves element-set memberships as *set-typed data*. The data can also encompass additional attributes of the elements or the sets. In the following, we give examples of how set-typed data are represented and what special cases, specific features, similarity measures, and tasks are associated with them.



|A|	16
|B|	16
|C|	12
|A∩B|	4
|A∩C|	4
|B∩C|	3
|A∩B∩C|	2

(a)

Paper Title	Year	ACM Classes
Is a bot at the controls?: Detecting inpu	2007	C
Seven at one stroke: results from a cach	2006	D, B, E, F
Wildlife net-gamekeepers using sensor	2007	K, C
ARMA(1,1) modeling of Quake4 Server t	2007	C
Adaptive &Delta;-causality control with	2007	H, K
An immersive voice over IP service to w	2007	C, G

(b)

Woman	Club
EVELYN	c1
EVELYN	c2
ELEANOR	c5
THERESA	c2
THERESA	c3
KATHERINE	c14
SYLVIA	c7

(c)

Movie Title	Year	Action	Comedy	Drama	Crime	Thrille
Toy Story	1995	0	1	0	0	0
Jumanji	1995	0	0	0	0	0
Grumpier Old Men	1995	0	1	0	0	0
Waiting to Exhale	1995	0	1	1	0	0
Heat	1995	1	0	0	1	1
Sabrina	1995	0	1	0	0	0
Tom and Huck	1995	0	0	0	0	0
Sudden Death	1995	1	0	0	0	0
GoldenEye	1995	1	0	0	0	1

(d)

Figure 2: Various forms of set-typed data: (a) the cardinality of set relations, (b) a multi-valued attribute (in grey), (c) a membership list, (d) Boolean attributes (in grey).

2.1. Data Representation

There are several ways to represent a set family on the data level, depending on the information available. One way is to explicitly represent the relations between the sets in the family. The data stores the absolute or relative size of the intersection of these sets (Fig. 2a). This representation does not require information about the individual set elements, and is hence suited when this information is unavailable or when dealing with infinite sets. For example, when the sets represent events, relative sizes can be used describe joint probabilities for these events. Also, when the sets represent logical variables, the data can involve infinite sizes.

When the number of elements in the set family is finite and their set memberships are available, three data structures for graphs can be used to represent these memberships. A multi-valued attribute can specify the sets to which each element belongs (Fig. 2b), resembling adjacency lists. Alternatively, a table of element-set memberships can be used (Fig. 2c), resembling an edge list. Boolean attributes representing the sets can also be used to specify which elements belong to them (Fig. 2d), resembling an adjacency matrix. These representations illustrate a duality between the elements and the sets: by transposing the matrix, each set S can be treated as an element that belongs to the dual sets corresponding to the elements of S. Also, adjacency lists can represent element lists as in the extensional definition of sets.

Apart from set membership, further attributes of the element might need to be involved in the analysis. For example information about club members might encompass their age and sex. A special type of attribute is associated with the set membership, for example the membership date in each club. This means for each set, one such attribute is needed to store the set-dependent values. Certain techniques support visualizing set-dependent attributes (Sect. 5.1).

An important notion in a family of sets is the *set membership degree* of an element, which we refer to as the element's degree. This degree denotes the number of sets in the family the element belongs to. It corresponds to set cardinality in the dual representation of sets and elements. A related concept is the *exclusive membership* to certain sets or set intersections. Many analysis tasks and visualization designs are concerned with the degree of certain elements and with exclusive membership, as we explain in Sect. 3 and Sect. 4.

2.2. Scope and Special Cases

In general, the sets in a set family overlap, i.e., they have one or more intersection relations. When all sets are in exclusion relation, they exhibit no overlap and define groupings over the respective elements. If such sets cover all the elements, they define a classification of the elements into classes (also called partitioning). In such cases, the set memberships can be represented by one categorical attribute that stores these classes. When the sets exhibit both exclusion and inclusion relations, but no intersections, they define a hierarchy over their elements. We limit our survey to techniques for visualizing overlapping sets. Hence, visualizing hierarchies and non-overlapping groups is not covered in this report.

A family of sets defined over a finite number of elements is equivalent to a hypergraph whose hyperedges represent the sets. A hypergraph is usually drawn either in subset standard (Sect. 4.3.1) or in edge standard (Sect. 4.4) [Mäk90]. For more details about available drawing techniques, refer to graph drawing literature [BCPS12, BVKM*10, KvKS09].

In some cases, there are constraints on possible intersection relations between the sets. One example is when an element can belong to a maximum of $k < m$ sets from a family of m sets. Another example is when a set can intersect with k other sets at most. It is important to identify and exploit such special cases, as some of the techniques presented in Sect. 4 can be simplified under such assumptions.

2.3. Similarity Measures

Many tasks related to set-typed data are concerned with finding which pairs of sets S_1 and S_2 exhibit higher similarity than other pairs, with regard to the number of shared elements between them $|S_1 \cap S_2|$. Several similarity measures between finite sets have been proposed in the literature. A symmetric measure was proposed by Jaccard [HHH*89]:

$$Jaccard(S_1, S_2) = |S_1 \cap S_2| / |S_1 \cup S_2|$$

It has been employed in set visualization both explicitly to reveal set similarity as in Radial Sets (Fig. 1c) and implicitly for matrix reordering (Sect. 4.5). Tversky [Tve77] proposed a generalized index for set similarity that can replicate other measures by using different parameterizations. It is also possible to weigh shared elements differently when computing the similarity. For example, elements of degree 2 in $S_1 \cap S_2$ can be weighed higher than other elements, as they belong exclusively to the overlap. An important issue with similarity measures is their sensitivity to the respective set sizes. Larger sets have higher probability of overlap, causing a bias in the above-mentioned measures. Applying the χ^2 statistic can eliminate such bias [AAMH13].

The choice of an appropriate similarity measure depends on the data, the tasks to be solved and the information to be communicated by the visualization. Depending on whether the chosen measure is symmetric or not, and on the value range it takes (e.g. $[0,1]$ or $[-1,1]$), different visual variables are appropriate for encoding set similarity, such as size [KSB*09], colour [AAMH13], position [LLS05] or order [KLS07]. Additionally, some techniques compute element-element similarities based on their set memberships, e.g. to optimize element ordering [KLS07].

3. Common Tasks with Set-typed Data

When designing a visualization of set-typed data, it is important to determine which tasks it needs to support. Here we list general tasks addressed by the surveyed techniques, classified into the following categories.

3.1. Tasks related to elements

These tasks are concerned with the membership of the elements in the sets.

A1 Find/Select elements that belong 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.

3.2. Tasks related to sets and set relations

These tasks are concerned with higher-level reasoning about the sets without taking individual elements into account. Example tasks applied to sets A, B and C include:

B1 Find out the number of sets in the set family.
B2 Analyze 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 Analyze inclusion hierarchies: e.g. find out if A is included in B, and B in turn is included in C (and so on).
B4 Analyze exclusion relations: e.g. find out if A does not intersect B, or $B \cap C$, or $B \cup C$.
B5 Analyze 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 involved in a certain intersection.
B8 Identify set intersections belonging to a specific set.
B9 Identify the set with the largest / smallest number of pairwise set intersections.
B10 Analyze 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 Analyze and compare set similarities: e.g. find out which pairs of sets exhibit high or low similarity according to some similarity measure.
B12 Analyze 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 de-emphasize the remaining data.
B14 Create a new set using set-theoretic operation: e.g. create the complement of A, or $A \setminus B$ as a new set to compare with other sets.

3.3. Tasks related to element attributes

Set-typed data can encompass additional attributes of the elements. The following tasks are concerned with how the element memberships and attributes are inter-related.

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 (Sect. 4.3). In this case, the task supports estimating the spatial distribution of a set [DvKSW12].
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 Analyze 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. Shneiderman emphasized the importance of supporting such queries in his task taxonomy [Shn96] and the role of set-theoretic operations to combine multiple constraints on the attribute values.

In the next section we survey state-of-the-art techniques that address the generic tasks listed above. A number of other tasks are also concerned with set-typed data such as hierarchical clustering of the sets or the elements, comparing multiple instances of a set family, and analyzing changes in the data over time. Such tasks are often application-specific and require dedicated techniques, and hence are not addressed explicitly in this survey.

4. The Survey

We conducted this survey by examining the titles published in the main visualization conferences and journals as listed in the supplementary material. After identifying relevant articles, we extended the search to further articles citing them in other venues. We classified the techniques we found into seven categories according to the main visual representation they use for depicting set relations. The techniques in each category exhibit similar scalability and readability properties as well as design considerations. Also, certain tasks are better supported by a certain category of techniques as we discuss in Sect. 5. The following subsections list the seven visual categories and describe the techniques in each of them. Certain techniques, however, might belong to multiple categories. Links to available software implementations or demonstrations and to additional resources about these techniques are available in the supplementary material and in the companion website http://www.setviz.net.

4.1. Euler and Venn Diagrams

Euler and Venn diagrams are amongst the oldest [Bar69] and most popular set visualizations. Sets are represented by labelled closed curves (of various shapes e.g. circles, ellipses or polygons) and set relations are depicted by the curve overlaps. Any set inclusion, exclusion and intersection can be represented with an Euler diagram as there are no restrictions on how the curves overlap. A Venn diagram is a restricted form of an Euler diagram as it has to show all possible combinations of curve overlaps. Thus, Venn diagrams quickly become visually complex as more sets are depicted.

The visual properties of Euler diagrams are simple yet perceptually powerful for depicting set relations [War12]. The closed curves clearly indicate set membership, as the perceptual tendency to organize space into regions is much stronger when indicated by closed curves than by proximity or similarity [Pal92]. The set relations are also easily visible, as the closed curves pop out preattentively, particularly when smooth [TS85, Kof35].

Euler diagrams were originally used to teach categorical propositions and syllogisms [Eul68]. They are still used to teach set theory, but are now also used in areas such as genetics and proteomics [MM11, RGBS*11] and reasoning systems [Sta05]. An Euler diagram is *well-matched* to what it represents when the spatial relationships between the curves precisely reflect the set relations. Euler diagrams are most effective when they are well-matched [Gur99], however, this cannot always be achieved without less effective aesthetics.

Various automatic drawing techniques that generate Euler diagrams with different aesthetic features and for different types of data have been developed. We provide an overview of implemented techniques in Table 1 and we discuss these further in the next sub-sections (see surveys on Venn [RW97] and Euler [Rod13] diagrams for more details).

Table 1: Features of implemented automatic drawing techniques for Euler and Venn diagrams.

for any relation	# of curves	well-matched	well-formed	smooth curves	curve shape	symmetric curves	region shading	area-proportional	cardinality glyphs	example techniques
✓	any	✓			polygon					[SRHZ11,SAA09]
✓	any	✓		✓	circle	✓				[SFRH12] (no shading)
✓	any		✓	✓	circle	✓	✓			[SFRH12] (shaded)
	any	✓	✓		polygon					[FH02]
	3	✓	✓	✓	circle	✓		✓		[LM13,CR05a]
✓	any			✓	circle	✓		✓		[Wil12]
	3	✓	✓	✓	ellipse	✓		✓		[MR14]
✓	any				polygon	✓		✓		[KMK*08]
	3	✓			polygon			✓		[RFSH10,CR03]
✓	any	✓			polygon			✓		[CR05b]
✓	1-3	✓			circle polygon			✓		[RHSF14]
	3		✓	✓	circle	✓			✓	[Cla08]
	3			✓	ellipse	✓			✓	[MDF12]

4.1.1. Well-formedness and Aesthetics

Studies indicate that the layout of the Euler diagram and its aesthetics affect user comprehension. An effective Euler diagram should be well-formed [RZP12], as in Fig. 3a. A *well-formed* Euler diagram has: simple curves that meet at most at one point in which case the curves cross; every set is represented by at most one curve; every set relation is represented by at most one region. Diagrams with concurrent curves or more than one curve for a set or more than one region for a set relation are the least effective for human comprehension [RZP12].

Euler diagrams with non-smooth curves or curves that are close to one another impede understanding [BR07]. Those drawn with circles are the most effective, but if circles cannot be used, the curves should be highly symmetric and the shape of the regions should be highly distinguishable from that of the curves [BSR*13]. Nevertheless, well-matchedness can be more important than well-formedness [CSR*14].

However, as illustrated in Table 1 it is not always possible for a drawing technique to satisfy all of these aesthetic criteria. This often depends on whether the technique draws diagrams for any or for only specific types of set relations.

4.1.2. Techniques for Any Set Relations

The techniques that draw well-matched diagrams for any set relations are often not well-formed and have non-smooth curves (e.g. [SRHZ11, SAA09, RZF08]), as in Fig. 3a. The smoothness, shape and closeness of the curves of the generated diagram could be further improved by other methods (e.g. [MR09, FRM03]), as shown in Fig. 4, but not well-formed diagrams are likely to remain not well-formed. Multiple curves representing the same set can be used to draw well-matched Euler diagrams with circles (e.g. [SFRH12]).

Techniques that draw well-formed Euler diagrams for any set relations often have smooth and highly symmetric curves like circles. However, the generated diagrams are not well-matched, as regions representing unwanted set relations are shown. The regions corresponding to these set relations are often shaded (e.g. [SFRH12, Ven80]), as in Fig. 3b, or left empty while glyphs are placed in the other regions (e.g. [MDF12, Cla08]), as in Fig. 6. However, shading was shown to be less effective than well-matchedness with respect to human accuracy and time [CSR*14].

4.1.3. Techniques for Specific Set Relations

A number of techniques generate a diagram for only those set relations for which a well-matched and well-formed Euler diagram can be drawn. No diagram is generated for other set relations. Thus when generated, the diagrams are more likely to have aesthetic features that aid comprehension, particularly when the curves are circles (e.g. [SZHR11]) rather than irregular polygons (e.g. [FH02]).

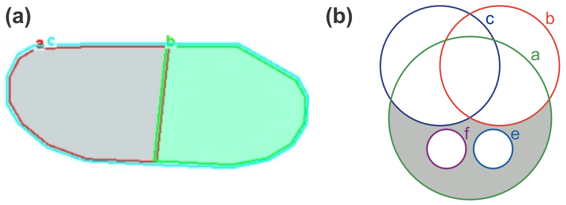


Figure 3: (a) A well-matched Euler diagram that is not well-formed and whose curves are not smooth [RZF08], and (b) a not well-matched Euler diagram with shading that is well-formed and has smooth curves [SFRH12].

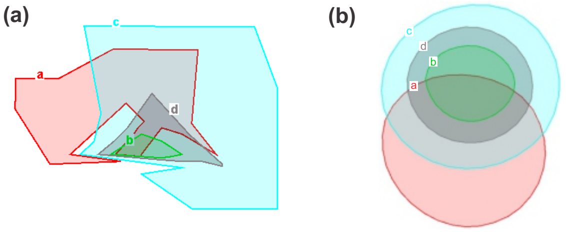


Figure 4: The layout improvement technique eulerForce [MR09] converts (a) to (b).

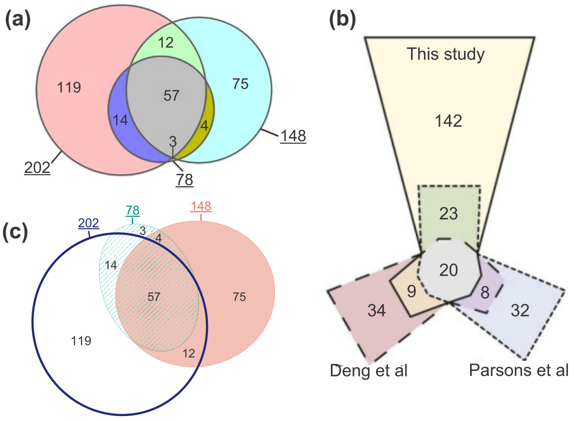


Figure 5: Area-proportional Venn diagrams drawn with: (a) circles [MM11] using Venn Diagram Plotter [LM13]; (b) polygons [RGBS*11] using 3 Circle [CR05b]; (c) ellipses using eulerAPE [MR14] for the numeric data in (a).

4.1.4. Techniques for Area-proportional Diagrams

Euler diagrams can be *area-proportional*, such that the area of each region in the diagram is directly proportional to the cardinality of the depicted set relation. Differences in these cardinalities are easily noted [TG80]. However, it is difficult and often impossible to draw accurate area-proportional diagrams with aesthetic features that aid comprehension.

Current techniques differ mainly in the shape used for the curves. Most techniques use circles (e.g. [LM13, Wil12, CR05a]) to facilitate user comprehension. However, circles have limited degrees of freedom and so, the generated diagrams are less likely to have accurate region areas. This is particularly problematic for Venn diagrams, as an accurate area-proportional Venn diagram can be drawn for any data with only two sets [Cho07]. Most circle-based techniques can produce misleading diagrams, e.g. in Fig. 5a, the region with 3 is much smaller than that with 4.

Other techniques use polygons (e.g. rectilinear curves [CR03], 4- or 5-sided convex curves [RFSH10], irregular curves [CR05b]) to generate accurate diagrams for most data. However, these diagrams are often difficult to comprehend, as they are not well-formed and have non-smooth and non-symmetric curves, as in Fig. 5b. Techniques that use regular polygons (e.g. [KMK*08]) produce symmetric curves, but have the same limitations as those using circles.

A recent technique, eulerAPE, [MR14] uses ellipses. Ellipses are smooth like circles, but have two more degrees of freedom. This means, the generated diagrams are more likely to be accurate and aesthetically desirable, as demonstrated by eulerAPE's evaluation for 3-set data, and Fig. 5c.

Current techniques often draw diagrams with only two or three curves and do not allow any regions with zero area. Exceptions include: venneuler [Wil12], which draws often inaccurate diagrams with any number of curves using circles; Rodgers et al.'s [RHSF14] technique, which draws accurate diagrams with up to three curves using a mix of circles and convex and non-convex polygons.

4.1.5. Techniques for Euler Diagrams with Glyphs

Humans are biased to area judgement [CM84]. Hence, techniques have been devised to generate Euler diagrams with *glyphs*, such that glyphs (not area) indicate the cardinality of the set relations, while the curves depict the set relations.

Equally-sized glyphs that are directly proportional in number to the cardinality of the set relations are typically placed in the corresponding regions of the diagram. TwitterVenn [Cla08] draws such diagrams to depict the number of twitter messages that used any of two or three user-selected words (Fig. 6a). eulerGlyphs [MDF12] draws similar diagrams with randomly or uniformly positioned glyphs and curves that are either area-proportional or not for Bayesian problems to reduce fallacious reasoning (e.g. Fig. 6b-c).

Differently-sized and multi-attribute glyphs can be used to depict different associated quantities [Bra12] (Fig. 6d), but no automatic drawing techniques have been devised.

4.1.6. Other Techniques

A technique that draws 3D Venn and Euler diagrams was recently introduced [RFS12]. Sketching software SketchSet [WPS*11] and SpiderSketch [SDRP11] were also devised to respectively generate Euler diagrams and Euler diagrams with graphs or shading from hand drawn sketches. Techniques that draw Euler diagrams for diagrammatic reasoning systems [Sta05] (e.g. spider diagrams [HST05], constraint diagrams [SD08]) and interactive diagrammatic theorem provers [UJ12, UJSF12] are also available.

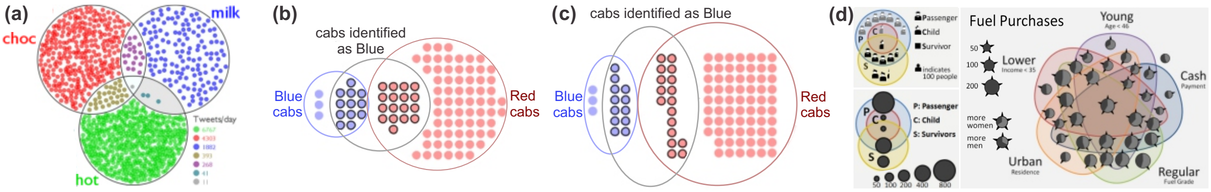


Figure 6: Euler diagrams with glyphs: (a) TwitterVenn [Cla08], (b)-(c) eulerGlyphs [MDF12], (d) Brath's [Bra12].

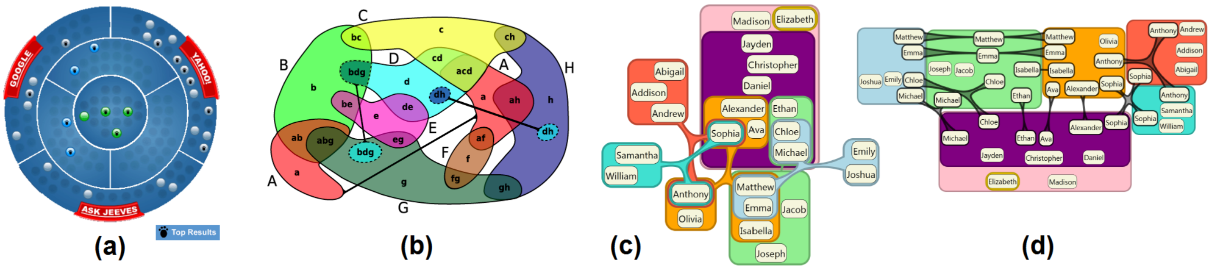


Figure 7: Euler diagram variants: (a) Missing Pieces [KSJ*06], (b) visualizing undrawable Euler diagrams [SA08], (c, d) untangling Euler diagrams [HRD10] using set splitting (c), and element duplication (d).

4.1.7. Diagram Design

Different designs have been used to draw Euler diagrams, but no empirical studies have been conducted to assess their effectiveness.

A different colour per curve is often used. If the curve interior is not coloured (e.g. Fig. 3b), the curves in which a region is located might not be easily identified. If the curve interior is coloured, transparency can be used (e.g. Fig. 3a, Fig. 4). However, the colours of the curves could perceptually fuse at overlaps and the colours of regions in the same curve could seem unrelated, giving the impression that they belong to different sets. The same issue is evident when a different colour per region is used (e.g. Fig. 5a), irrespective of whether the colours of regions in the same curve are somehow related (e.g. Fig. 5b). To avoid such problems a weaving approach [LRS10] has been proposed, so a diagram like Fig. 8b is drawn instead of Fig. 8a.

In contrast to other drawing techniques, eulerAPE (Sect. 4.1.4) draws the curves using different visual feature channels (namely colour, outline and texture, as in Fig. 5c) so the curves are visually distinct and do not fuse perceptually. The curves in which the specific regions are located are thus easily identified. Also, by tuning one's attention to the feature type of a curve, other feature types recede and one can better focus on a specific curve representing a set [War12].

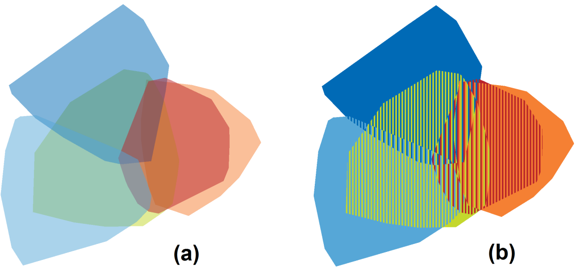


Figure 8: An Euler diagram filled using: (a) transparency, (b) weaving. [LRS10]

4.2. Euler Diagram Variants

Several variations of Euler diagrams have been proposed for different purposes. Like Euler diagrams, these techniques use closed regions to represent the sets or subsets thereof.

Missing Pieces [KSJ*06] use concentric rings for showing the results of three search engines (Fig. 7a). The outer and middle rings include the elements retrieved by one or two engines, respectively. The inner ring includes elements retrieved by all three engines. The search results are represented as glyphs inside the respective regions and can be coloured to encode additional attributes. *Fan diagrams* [KLS07] use a similar layout to visualize three sets. Instead of having a separate ring for pair-wise overlaps, these overlaps are placed between the respective parts in the outer ring.

To handle cases where Euler diagrams cannot be drawn, Simonetto et al. [SA08] proposed a drawing method based on the corresponding Euler graph. Set relations that cannot be represented in a proper Euler diagram are visualized by splitting or duplicating certain sets and subsets into disjoint parts, and connecting these parts using edges (Fig. 7b).

Similar ideas were proposed to *untangle* Euler diagrams and ensure their drawability using simple rectangular shapes [HRD10]. Two variations, called *ComED* and *DupED* use set splitting and element duplication respectively. *ComED* splits a set into multiple rectangular parts, depending on how it overlaps with larger sets (Fig. 7c). These parts are connected with hyperedges that preserve the continuity of the set regions, as in Euler diagrams. However, the hyperedges contain no elements and hence their mutual crossings represent no shared elements between the respective sets. The rectangular parts are arranged in a containment hierarchy that reveals several set relations. For example, in Fig. 7c it is evident that all elements shared between the blue and the pink sets also belong to the green and purple sets. *DupED* creates separate rectangular regions for the sets, and duplicates the elements that belong to multiple sets. Multiple instances of the same element are linked with hyperedges (Fig. 7d). It outperforms *ComED* in counting the sets, comparing their sizes, and assessing their intersections. However, *ComED* scales significantly better in terms of visual complexity.

4.3. Overlays

In many data analysis scenarios, the set memberships are a secondary information in the data that needs to be analyzed in the context of other data features. For example, when the elements have a spatial reference, they are often viewed on a map that provides context information about their locations. Other examples are points in a scatter plot or nodes in a graph. Several techniques have been proposed to augment set memberships over the elements in an existing visualizations. These techniques can be classified into the following three categories, according to the visual elements they use.

4.3.1. Region-based Overlay Techniques

These techniques surround the elements of a set with a closed curve that defines a region. One element can belong to multiple regions if it belongs to multiple sets.

Bubble Sets [CPC09] constructs a contour (also named implicit surface) for every set so that it includes all of its elements and excludes all other elements if possible. For this purpose it computes an energy map over the pixels in the convex hull containing the set elements. In a second step, it applies the marching squares algorithm to compute the implicit surface from the map. The sets are assigned semi-transparent colours to reveal their overlaps and to keep the context visualization visible. Unlike Euler diagrams, two regions might overlap even if their sets share no elements. Such overlaps should be understood as artifacts that encode no information. An *inverse distance-based potential field* [VPF*14] alleviates these artifacts but might result in disconnected regions. Bubble sets were demonstrated to overlay set memberships over tens of elements in a scatter plot, a graph, or a map (Fig. 1b and Fig. 9a). Depending on the overlap extent, the technique can handle between 4 to 20 sets and still retains enough visibility of the context.

Texture splatting has been proposed to depict areas of interest (AOIs) in software architecture diagrams [BT06]. Splatting is applied to a skeleton constructed from the diagram elements according to their size and position. A post-processing step erases elements that incorrectly fall within a specific AOI. Overlaps between multiple AOIs are emphasized using subtractive colour blending which creates darker overlapping regions (Fig. 9b). A follow-up work [BT09] uses texture and colour to encode further software metrics about AOI elements. Splatting creates smooth boundaries and is applied there only, as it is computationally exhaustive.

When the base visualization places elements that belong to the same set close together, simpler and more convex shapes can be used for the regions than in previous techniques. A typical case is when the sets indicate clustering results of a graph. *Vizster* [HB05] create a region for each cluster by computing the convex hull of the nodes in it and interpolating the hull boundaries using a cardinal spline (Fig. 9c). If the clustering algorithm allows node membership to multiple clusters, the cluster overlaps are revealed by colour.

4.3.2. Line-based Overlay Techniques

To reduce the ink used in the overlay and the interference with the base visualization, many techniques use lines to represent set membership. Elements that belong to the same set are shown by being present on one or more connected lines.

LineSets [AHRRC11] computes a line for each set that passes through its elements (Fig. 10a) using a travelling salesman heuristic that minimizes the line length. This in turn reduces self-crossings and bends, making it easier to follow the line. The lines are drawn as piecewise Bézier splines of different colours. As with region-based methods, not all line crossings represent set overlaps. Actual overlaps are marked with concentric rings around the elements colour-coded according to the respective sets. Interaction makes certain lines salient, while the other lines are drawn thinner to reduce visual clutter. LineSets were shown to scale better than region-based methods and can overlay up to tens of sets over hundreds of elements. However, the use of simple lines imposes an artificial ordering on the set elements.

Kelp Diagrams [DvKSW12] connects the elements in a set using a graph structure instead of a simple line. It surrounds each element with a circle clipped to its Voronoi cell to avoid overlaps. Then it computes a tangent visibility graph based on these clipped circles. Finally, it constructs for each set a minimum cost path as a subgraph that connects its elements. This graph aims to capture the shape of a point set on a map. The links are routed so that no path contains elements that do not belong to the respective set. Two styles were proposed to draw overlapping links. The nested style draws the links over each other, with thinner links in the top to ensure their visibility (Fig. 10c). The striped style uses alternating stripes for areas that contain elements of multiple sets (Fig. 11c). A follow-up technique called *KelpFusion* [MHRS*13] allows the graph to vary from a minimum spanning tree to the convex hull of a point set. It uses a hybrid representation that bridges Bubble Sets (Sect. 4.3.1) and Kelp Diagrams using both lines and regions (Fig. 10d).

In some cases, the base visualization represents the elements of each set separately, and hence creates multiple instances of the same element. An example of this are the *parallel tag clouds* [CVW09] that represent multiple sets of tags (Fig. 10b). This technique connects multiple instances of the same tag with a thick path line. To avoid clutter, only the two ends of the edge connecting a tag instance with its next occurrence are depicted. The full segment is shown only for selected tags on demand. While it is hard to follow the instances of an unselected tag, the depicted edge ends reveal if such instances exist or not in parallel clouds.

The *context-preserving visual links* [SWS*11] are a generic technique that uses line overlays to link multiple instances of the same element in multiple coordinated views showing different visualizations. The layout algorithm routes the lines preferably within white space using a density map to minimize interference with the base visualizations.

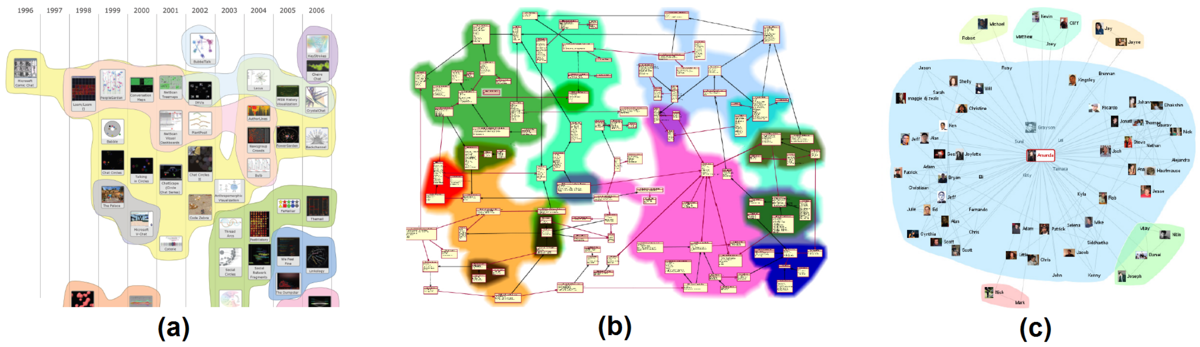


Figure 9: Region-based overlay techniques: (a) Bubble Sets showing groups of items over a timeline [CPC09], (b) texture splatting to depict areas of interest [BT06], (c) convex hulls to depict clustering results of a social network in Vizster [HB05].

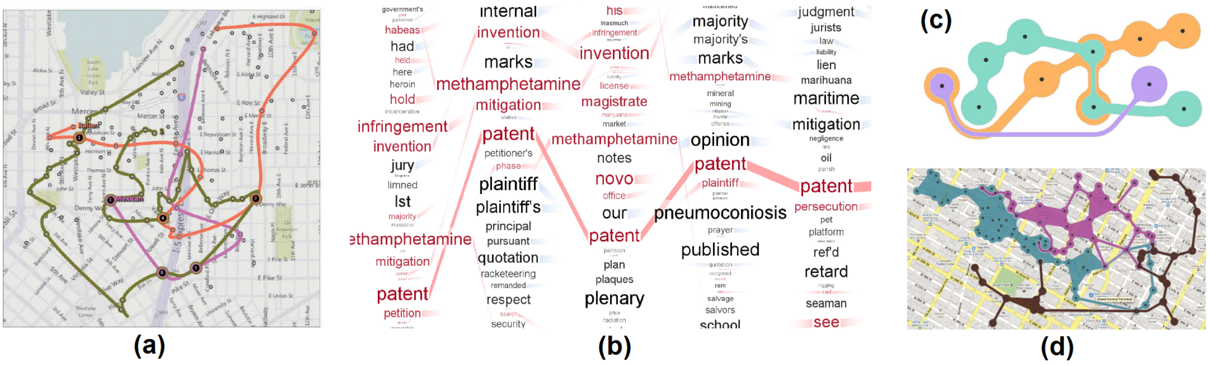
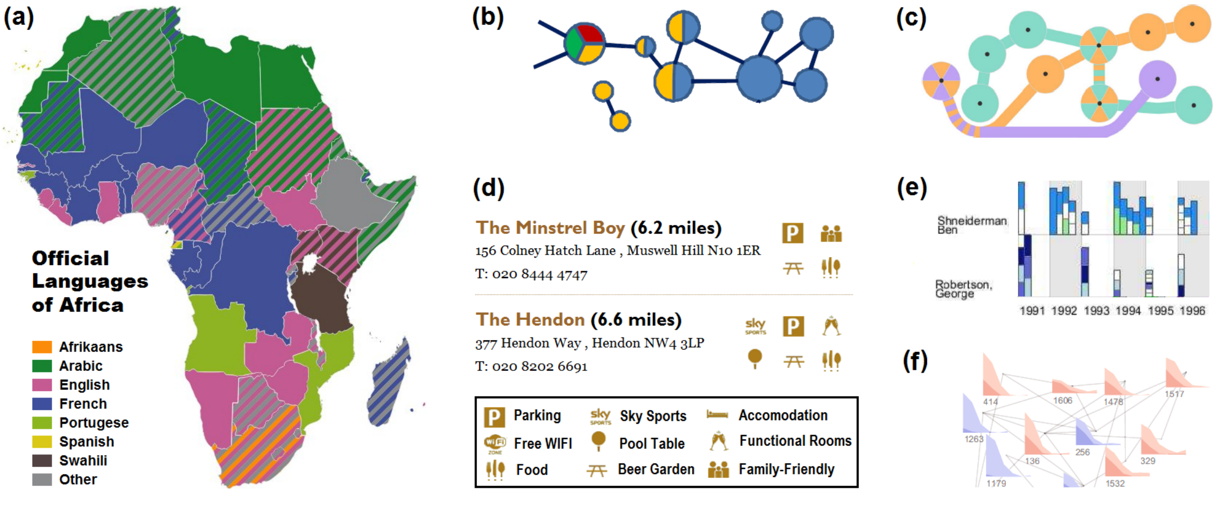


Figure 10: Line-based overlay techniques: (a) LineSets [AHRRC11], (b) Parallel Tag Clouds [CVW09], (c) Kelp Diagrams [DvKSW12], (d) KelpFusion [MHRS*13] uses a hybrid region- and line-based representation.



The Minstrel Boy (6.2 miles)
156 Colney Hatch Lane , Muswell Hill N10 1ER
T: 020 8444 4747

The Hendon (6.6 miles)
377 Hendon Way , Hendon NW4 3LP
T: 020 8202 6691

Parking
Sky Sports
Accomodation
Free WIFI
Pool Table
Functional Rooms
Food
Beer Garden
Family-Friendly

Figure 11: Glyph-based overlay techniques: (a) multi-colour hatching of a map [Wik10], (b) colour-coded nodes [IMMS09], (c) striped glyphs in Kelp Diagrams [DvKSW12]. (d) using icons [Kin], (e) colour-coded bars [SOTM06] to indicate the co-authors of each paper, (f) glyphs [XDC*13] to indicate correlations between the graph structure and set memberships.

4.3.3. Glyph-based Overlay Techniques

In many cases, it is enough to represent the set memberships for the individual elements in the base visualization, without the need to represent each of the sets as a connected object. In this case, glyphs can be used as simple overlays.

Coloured glyphs are commonly used for this purpose: each set is assigned a colour from a qualitative (categorical) colour scale. There are several ways to design coloured glyphs to show multiple set memberships, depending on how the base visualization represents the elements. For example if the element's representation has enough space, multiple coloured-coded dots can be added in an appropriate location to indicate its set memberships. When the elements are regions on a map, hatching techniques can be used to fill these regions with multiple colours (Fig. 11a). Pie-like glyphs are commonly used to overlay set memberships when the elements are represented as circles, such as the nodes of a graph [IMMS09] (Fig. 11b). *BiblioViz* [SOTM06] represents papers as bars in a time line, and overlays coloured segments over the bars to represent multiple co-authors (Fig. 11e). However, the division of a circle or a bar into coloured segments might causes a bias regarding the order and size of these segments and a confusion with the spatial layout. *Kelp Diagrams* provide an alternating pattern to avoid this confusion and minimize discontinuity (Fig. 11c).

Colour composition [HKvK*13] is another way to indicate multiple set memberships. For example purple can be used to indicate membership of both red and blue groups. However, this is restricted to two or three sets, as it is otherwise hard to memorize all possible colour compositions.

The use of colour can support the inference of the spatial distribution of the sets (Fig. 11a). Instead, the set memberships can be indicated using icons (Fig. 11d). This is appropriate when the sets represent real-world concepts that have corresponding icons such as flags or common signs. However, without interaction, a serial scan might be needed to find out which elements belong to which set.

Other types of glyphs have also been devised for specific applications. For example, glyphs based on superimposed area charts [XDC*13] were proposed to encode how the distance between two nodes in a graph correlates with the number of set memberships shared between them (Fig. 11f). Also, *MetaCrystal* [Spo04] uses polygonal glyphs to represent meta search results. The number of sides encodes how many search engines retrieved a specific document, with multiple colours encoding these engines. Finally, coloured pie-like glyphs were proposed to visualize fuzzy membership of overlapping communities in networks [VRW13].

Overlay techniques allow the analysis of how certain information and relations between the elements correlate with their set memberships. Alternatively, these correlations can be augmented with other visualizations that better emphasize the set information as in some of the following techniques.

4.4. Node-link Diagrams

The membership relations between elements and sets can be modelled as edges of a bipartite graph. Several techniques have been proposed to visualize bipartite graphs.

A simple layout for bipartite graphs places the elements and the sets in two lists parallel to each other. *Jigsaw* [SGL08] uses this layout to show co-occurrence relations between different concepts in documents (Fig. 12a). Schulz et al. [SJUS08] demonstrated techniques to reduce the clutter caused by crossing edges in such layouts using colour blending and a fisheye lens. Both systems show how additional attributes of the elements can be depicted using colour or additional columns.

Anchored maps [Mis06] use a circular layout to visualize bipartite graphs. The technique depicts the set nodes around a circle, and element nodes as free nodes depending on their set memberships (Fig. 12b). Elements that belong exclusively to a set are placed as a bundle of nodes outside the circle, originating from the respective set node. Elements that are shared between multiple sets are placed within the circle, depending on their set memberships.

PivotPaths [DHRRD12] is designed to support strolling in multi-faceted information spaces. Its node-link layout can also be used to depict element-set memberships, by placing the set nodes in the middle line (Fig. 12c). An element node is placed at a distance from the middle line that is proportional to its set membership degree. Its horizontal position is computed as the mean of the set nodes it is connected to. The elements can be divided into two groups and placed at different sides of the middle line.

Node-link diagrams can also be used to show the similarity between the sets as links between respective nodes. *Circos* [KSB*09] uses a circular layout for the nodes, and stripes of varying thickness to connect the nodes. The stripe thickness encodes the number of elements that fall in both categories. *Radial Sets* [AAMH13] use a similar metaphor to encode set overlaps (Sect. 4.6 and Fig 15). Unlike Circos, the links originate from the same location, to emphasize that the elements in a certain overlap between two sets can also belong to other sets and overlaps.

4.5. Matrix-based Techniques

Different methods have been proposed to visualize set memberships using matrices. These approaches take advantage of the clear and flexible metaphor of matrices.

ConSet [KLS07] maps sets and elements to rows and columns respectively. The cells encode set memberships (Fig. 13a). The rows and columns are reorderable, as set and element names have no predefined order. The reordering can both simplify the matrix and reveal patterns in it, such as clusters of elements that exhibit similar set memberships. Several interactions and visual aids are possible with the

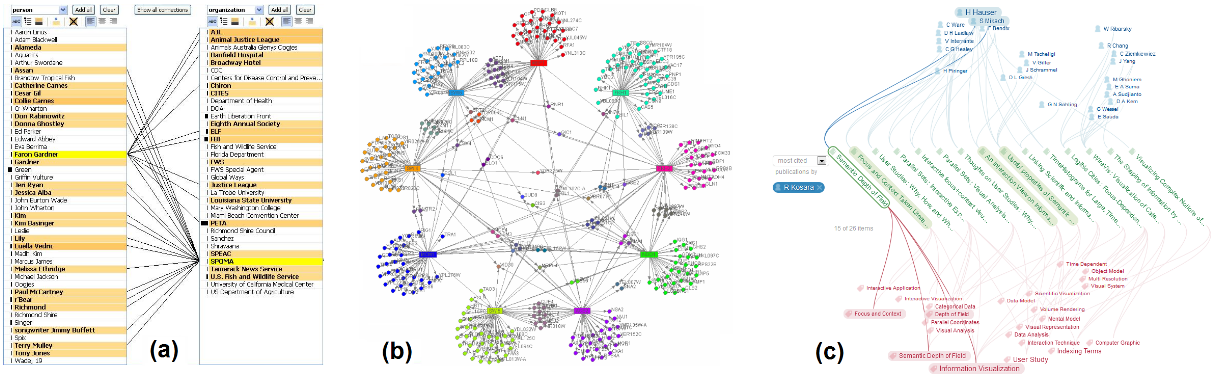


Figure 12: Node-link techniques: (a) Jigsaw [SGL08], (b) anchored maps [Mis06], (c) PivotPaths [DHRRD12].

matrix representation, such as the aggregation of elements or sets. Aggregated elements can be indicated visually using darker cells or additional bars. To facilitate inferring to which sets an element belongs, the cells can be coloured by unique set colours. Also, to facilitate inferring the elements that belong to a set, the respective cells can be connected with a line, instead of showing grid lines [ZKBS02].

PixelLayers [SDS13] represent each set as a separate matrix whose cells encode which elements belong to the set (Fig. 13b). Each element is represented by a unique cell position across all matrices. Hovering the mouse over an element highlights the respective pixels in the sets it belongs to. Drag and drop interactions allow aggregating multiple sets into one matrix using union or intersection (Fig. 13b).

Frequency grids [MDF12] represent the elements as cells in a matrix, and places a glyph in each cell to encode the respective set memberships (Fig. 13c). They facilitate element counting. However, they are limited to only a few overlap combinations between a small number of sets.

The techniques mentioned so far encode which individual elements belong to each set, and which ones do not. A matrix can alternatively depict how the sets overlap with each other, by representing the sets both as rows and as columns: Each cell contains a similarity measure between the respective sets (Sect. 2), encoded in colour as in a heatmap (Fig. 13d). Each pair of sets corresponds to two cells in the matrix. Therefore, the matrix can fit two symmetric measures, or one asymmetric measure. The matrix can be reordered to reveal clusters of sets that exhibit high overlap with each other. To explicitly represent the overlaps between triples of sets, each row (or column) can be divided recursively into multiple rows (or columns). However, the resulting matrix becomes difficult to comprehend and contains several redundancies, as each 3-set group is mapped to six separate cells.

KMVQL [Huo08] is a system to support formulating queries over a collection of items, by defining Boolean combinations of different search criteria. It encodes all possible 2^n membership combination of n sets in a matrix (Fig. 13e). The user can click on a cell to include the elements it represents in the query result. Also, the cells can be coloured to encode the frequency of elements in each combination of set memberships.

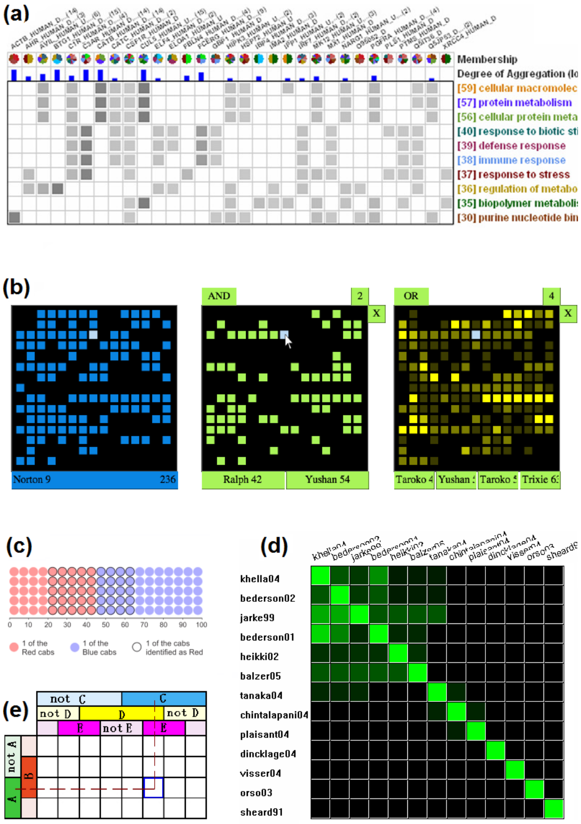


Figure 13: Matrix-based techniques: (a) ConSet [KLS07], (b) PixelLayer [SDS13], (c) frequency grid [MDF12], (d) similarity matrix [LLS05], (e) the KMVQL layout [Huo08].

4.6. Aggregation-based Techniques

When the number of elements is large, it becomes less feasible to depict and investigate how single elements belong to the sets. Following Shneiderman's visual information-seeking mantra [Shn96], many techniques provide an overview of such data first, and allow exploring details about certain elements on demand. These techniques employ frequency representations of set-typed data to show the number of elements in different sets and subsets. They aggregate multiple data elements into a single visual element that encodes this frequency.

Bar charts have been used to depict the sizes of the sets and reveal the set overlaps as the bars are brushed [AAMH13]. Unlike traditional bar charts, an element can be aggregated in multiple bars, as it might belong to multiple sets. Clicking on one bar selects the elements in the respective set, and highlights the fraction that these elements represent in the other bars, revealing how certain pairs of sets overlap (Fig.14a). The selection can be refined further using set operations between new selection and previously selected elements, to investigate the overlaps between multiple sets. However, this chart does not readily reveal how the sets overlap and can only depict certain overlaps on demand.

Set'o'gram [FMH08] is an extension to the interactive bar chart, designed for set-typed data. It divides the bars representing the sets into sections that correspond to elements of different degrees (Fig.14b). Starting from the bottom, the i^{th} section in a bar represents elements in the respective set that are shared with $i-1$ other sets. The height of a section is proportional to the number of elements aggregated in it. Starting from the top, the sections are assigned increasing widths and are shaded along their diagonals to distinguish between successive sections. The sections can be selected and highlighted individually for finer analysis on demand of the degree of overlaps.

Mosaic displays [Hof00] is a space-filling technique that recursively partitions the space along the categories of multiple categorical variables (Fig.14c). To visualize set-typed data, set memberships can be treated as binary categorical variables [FMH08]. However, using both horizontal and vertical subdivisions makes it hard to relate display tiles that belong to the same set.

The *Double-Decker plot* [HSW00] adapts mosaic displays to show how multiple Boolean variables correlate. It depicts how multiple sets overlap by partitioning the space according to the set memberships in the horizontal dimension only, showing how many elements belong to each possible combination of set memberships (Fig.14d). The partitioning hierarchy is depicted using an additional display which shows each set in a separate row using multiple tiles to represent its element. Starting from the bottom, row i is divided into 2^i parts that correspond to the different membership combinations of the sets $S_1..S_i$. This gives an overview of how the sets overlap, however, from the perspective of the set that defines the first partitioning level. In addition, the plot allows easy comparison between selected portions in different overlaps, as the respective tiles are of the same height. The *set co-occurrence view* [Wit10] uses a similar plot to support set-typed data in the bargrams interface. This interface uses additional rows to show the possible values of other attributes and the frequencies of these values. Kosara [Kos07] proposed a redesign of Venn diagrams composed of two parts, as with Double-Decker plot. The lower part of this redesign consists of a node-link visualization of the binary tree whose levels represent the memberships to the different sets. The upper part is a simple bar chart of the respective overlap sizes, allowing direct comparison of these sizes.

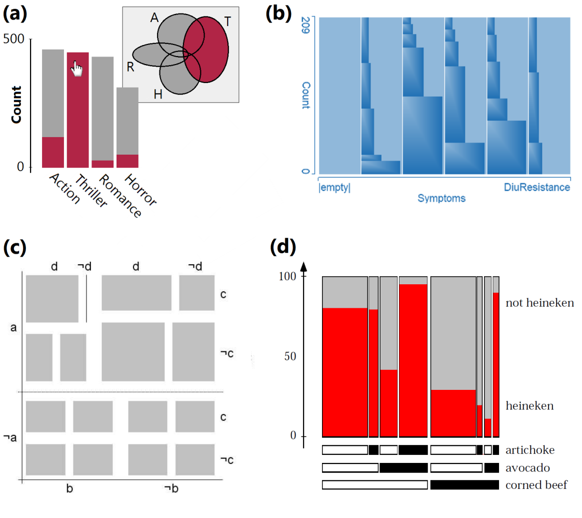


Figure 14: Aggregation-based techniques: (a) an interactive bar chart [AAMH13], (b) Set'o'gram [FMH08], (c) Mosaic displays [Hof00], (d) Double-Decker plot [HSW00].

Parallel Sets [KBH06] can be applied to visualize set-typed data by treating set memberships as binary categorical variables. Each set is represented on a separate horizontal axis using two boxes of proportional size to represent both the elements that belong to the set and the remaining elements. Up to four stripes connect the boxes between the two topmost axes to represent elements that fall in the respective set membership combinations. In the standard mode, the stripes are split further as they pass through the remaining axes, representing all possible set combinations. Unlike mosaic displays, Parallel Sets represent the elements of a set in one box only instead of several tiles. However, splitting the stripes increases them by a factor of 2, as with the mosaic tiles. Moreover, the stripes overlap, causing clutter with more than four sets. A bundled mode of the stripes reduces this clutter but causes stripe discontinuity.

Radial Sets [AAMH13] provide a more detailed overview of set-typed data than the above-mentioned techniques. The sets are depicted as non-overlapping regions with a radial ar-

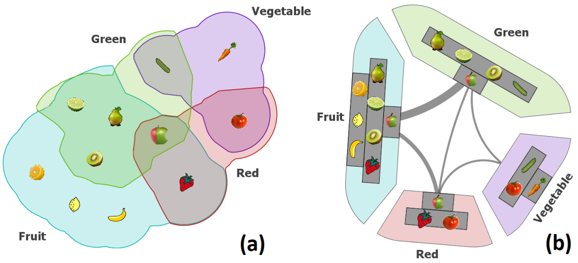


Figure 15: (a) An Euler diagram (adapted from [WWC09]), (b) equivalent Radial Sets [AAMH13] with illustrative icons. The histograms in grey show a breakdown of set elements by their degrees. The arcs show overlaps between pairs of sets.

rangement. The elements are represented as histogram bars inside these regions, grouped by their degrees (Fig. 15). Overlaps between pairs of sets are represented as links of proportional thicknesses. Overlaps between triples of sets are represented by hyperedges between the respective regions. A hyperedge is depicted as a node connected to the respective regions using tapered links. To avoid visual clutter, these links can be shown on demand by hovering over a node with the mouse. This results in a bubble chart of the overlaps which enables size comparison, but hinders the ability to visually infer which sets are involved in which overlap. Radial Sets use colour to indicate selected elements. When no elements are selected, colour can be used to encode aggregated attribute values or measurements of the elements aggregated in the histogram bars or the edges (Fig. 1c).

InfoCrystal [Spo93] uses glyphs to represent all possible set overlaps. The set labels are placed on a circle and act as magnets on the glyphs to determine their placement. A follow-up work [Spo04] demonstrates the use of glyph sizes to encode overlap sizes, and the use of colour to encode the sets involved in the overlap.

In some cases, there is a need to provide a compact overview of set sizes, as part of an information-dense interface. A common mistake is to show the set sizes via a pie chart, as the chart categories are not mutually exclusive and do not sum up as parts of a complete whole. Fan diagrams [KLS07] address this issue by explicitly visualizing the overlaps between the categories. An alternative way is to use stacked bars with a special motif [WMLP12] or along with an additional bar that indicates the actual number of the elements [BCH*13].

Except for mosaic displays (Fig.14c), the techniques mentioned above might represent one element in multiple visual elements, depending on the sets it belongs to. Some visualizations indicate this element redundancy explicitly, as with the links in Radial Sets and the collocated bars in Double Decker plots. Interaction is needed to investigate which elements are present in multiple sets, and to obtain detailed information about selected elements.

4.7. Scatterplot-based Techniques

One way to analyze similarity values between sets in detail is to use a 2D scatter plot that represents the sets as points in the plane. Though such a plot does not emphasize sets as containers of elements, it offers a clear layout to analyze the relations between the sets and identify clusters of similar sets. However, not all set similarity measures define a distance function, which limits the applicability of 2D projections (e.g. close points could be produced for disjoint sets).

The *scatter view* [LLS05] visualizes the similarity between a certain set, and the rest of the sets. It depicts two asymmetric similarity measures against each other to find which set is closer to the reference set both in overlap intensity and completeness (Fig. 16a). To gain an overview of the similarities between all pairs of sets, the authors proposed a *cluster view* that projects the sets on the plane similar to the way multi-dimensional scaling operates (Fig. 16b).

Correspondence analysis (CA) [Gre84] has been used to visualize 2-mode social networks by treating them as binary contingency tables [BH11]. Fig. 16c depicts the CA plot for the southern women dataset (Fig. 2c). The plot contains points both for sets and elements. Close element points indicate similar set memberships. Close set points indicate high overlap. Edges can be optionally overlaid between the sets and elements.

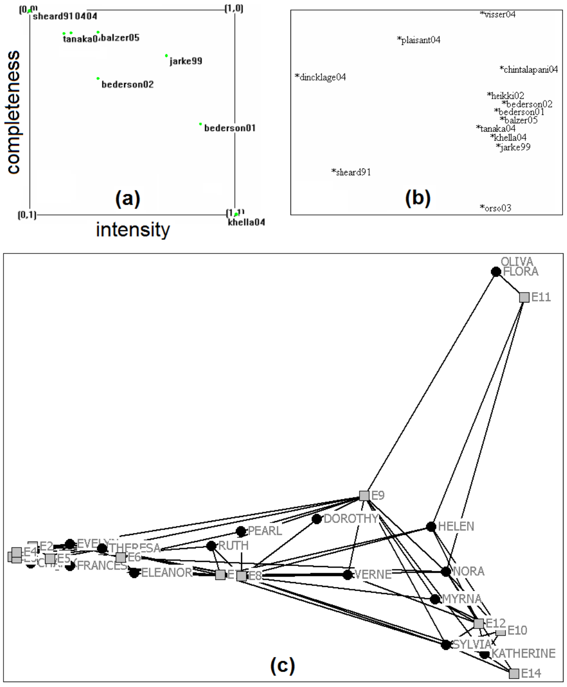


Figure 16: Scatterplot-based techniques: (a, b) a scatter view and a cluster view [LLS05], (c) Correspondence Analysis view of the southern women dataset [BH11].

5. Comparison and Findings

To provide guidance on applying the surveyed set visualization techniques to a given problem, we compare the techniques according to the following three aspects.

5.1. Comparison by what is represented

Set-typed data can encompass information about sets and their relations, elements and their set memberships, and other element attributes. The surveyed techniques differ by the type of information they represent:

- **Representing set information only:** These techniques provide no information about the individual elements. This includes simple Euler diagrams that represent set relations, as well as matrices, node-link diagrams, and scatter plots that represent set similarities.
- **Representing individual elements explicitly**: Examples are Euler diagrams with glyphs, overlays, element-set node-link diagrams, membership matrices and frequency grids. Further element attributes can often be represented using additional visual features or additional columns.
- **Representing element aggregates**: As discussed in Sect. 4.6, such techniques depict groups of elements, possibly along with relations between these groups. Some techniques (e.g. Double-Decker and Radial Sets) can also represent aggregated attribute values for group elements.

The techniques vary also in the set relations they represent explicitly. Euler diagrams show inclusion, exclusion, and intersection relations. Scatterplot-based and some aggregation-based techniques (e.g. Set'o'grams) do not represent these relations explicitly. Other aggregation-based, node-link, and matrix-based techniques represent certain set relations only (usually set intersections).

Finally, certain techniques show multiple instances of the same element according to the sets it belong to. Example for this are the DupED version of untangled Euler diagrams (Fig. 7d) and parallel tag clouds (Fig. 10b). Also, membership matrices fill multiple cells for the same element (Fig. 13a). Visual duplicates allow set-dependent attributes (Sect. 2.1) to be shown, e.g. different tag frequencies or ranks in multiple clouds.

5.2. Comparison of general strengths and weaknesses

Each of the techniques categories listed in Sect. 2 has advantages and limitations associated with the visual representation it employs. Table 2 summarizes the major ones that generally apply to the techniques in the respective category. However, it should be noted that individual techniques have their own advantages and limitations. For more details refer to Sect. 4 and to the respective articles.

5.3. Comparison by supported tasks and scalability

The surveyed techniques differ in the tasks (Sect. 3) they support. Table 3 provides an overview of the tasks supported by a representative subset of techniques from all surveyed categories. The task support was either indicated by the authors or judged by us based on published work. We indicate whether the task is supported fully, partially or through interaction only. Partial support means that the technique is not always effective for the respective task, or support the tasks to a limited extent (e.g. with few sets only). Additionally we give a rough estimate of the scalability of the techniques, both in the number of sets and in the number of elements, when applicable. Actual scalability limits depend on the complexity of the specific dataset, such as overlap strength and skewness in the set sizes.

Table 2: Selected strengths and weaknesses of the visual categories (Sect. 4). Euler diagram variants are not listed separately.

Category	Strengths	Weaknesses
Euler-based diagrams	Intuitive when well-matched (little training is required). Represent all standard set relations compactly.	Limited to few sets due to clutter and drawability issues. Desired properties not always possible (e.g. convexity).
Overlays	Emphasize element and set distributions according to other data features (e.g. map locations).	Often limited in the number of elements and sets. Undesired layout artifacts (overlaps, crossing, shapes, etc.).
Node-link diagrams	Visually emphasize the elements as individual objects. Show clusters of elements having similar set memberships.	Limited scalability due to edge crossings. No representation of set relations in element-set diagrams.
Matrix-based techniques	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.
Aggregation-based	Highly scalable in the number of elements. Some techniques can show how attributes correlate with set membership.	Usually, do not emphasize sets and elements as objects. Limited in the set relations they can represent.
Scatter plots	Show clusters of sets according to mutual similarity. Clutter free and scalable when showing sets only.	Do not represent standard set relations. Dots are often perceived as elements not as sets.

Table 3: Comparison of selected techniques from Sect. 4 by the tasks they support (Sect. 3). Selected Euler diagram variants are included in the first group.

Technique		Element-related Tasks							Set-related Tasks														Attribute-related Tasks					Scalability	
		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	in # of sets	in # of elements
Euler-based	Euler diagrams	●	●	●	○			○	○	●	●	●	●	○	●	●	○	●	○	○	n/a	○	○	○	○			about 10	hundreds / ∞
	ComED	●	●	●	○			●	○	●	○	○	○	●	○	○			○	○	●							10 to 20	hundreds
	DupED	●	○	○	○			●	●	●	●	●	○			○			○	○	●							about 10	tens
Overlays	BubbleSets	●	●	○	○			●	○	○	○	○	○	○		○	○			○	○		○	○				about 10	tens
	LineSets	●	●	○	○			●	○					○		○				○	○		○					10 to 100	hundreds
	Kelp diagrams	●	●	○	○			●	○	○	○	○	○	○	○	○	○		○	○	○		○	○				about 10	tens
	Colored glyphs	○	●	○	○				○				○	○		○				○	○					○		10 to 20	hundreds
	Icon lists		●					●												○			●			○	○	tens	large list
Node-link	Linked lists	👆	👆	👆	👆	👆	👆	👆	●												○		●	○	○	👆	👆	hundreds	hundreds
	Anchored maps	○	○		○		○		●				○		○	○		○		●	👆		👆					20 to 50	hundreds
	PivotPaths	👆	👆		👆		👆		●					○						👆	●		👆					50 to 100	hundreds
Matrix	ConSet	●	●	○		👆	👆	👆	●			👆	👆	○							👆	👆	●					about 100	about 100
	PixelLayer	●	👆	👆					●			👆	👆									👆	👆					tens	hundreds
	Frequency grids	●	●	○	○	○	○	○	○	●			●	●	●	●	○	●		●	●		👆					3 to 5	hundreds
	Overlap matrix								●			●	●			●	●	○	○	○	●							about 100	not applicable
	KMVQL	👆		👆					●			○	○	○	●	●					●				●			4 to 6	not applicable
Aggregation	Mosaic displays			👆					●			○	○	○	●	●		○		○	●				●			up to 4 sets	large (agg.)
	Double-Decker	○		●					●	👆	👆		●	○	●	●		●		●	●			●	●			4 to 6	large (agg)
	Sets’o’grams	👆		👆	●	👆	●		●				👆	○	👆	👆				○						👆	👆	50 to 100	large (agg.)
	Radial Sets	👆		👆	●	👆	●		●			●	●	●	👆	👆	●	●	●	●	●	👆		👆	●	👆	👆	20 to 30	large (agg.)
Scatter	Scatter view								○										○									hundreds	not applicable
	Cluster view								○										●									hundreds	not applicable

● Task is supported

○ Task is partially supported

👆 Task requires interaction

A1: Find/Select elements of a specific set
A2: Find sets containing a specific element
A3: Find/Select elements by set memberships
A4: Find/Select elements by their degrees
A5: Filter out elements by set memberships
A6: Filter out elements by their degrees
A7: Create a set out of certain elements

B1: Find the number of sets in a family
B2/3: Inclusion relations / hierarchies
B4/5: Exclusion / intersection relations
B6: Identify intersections between k sets
B7: identify sets involved in an overlap
B8: Identify intersections of a set

B9: Identify the set with largest / smallest number of pair-wise set intersections
B10: Analyze & compare cardinalities
B11: Analyze & compare set similarities
B12: Analyze & compare set exclusiveness
B13: Highlight specific sets, subsets, etc.
B14: create a set by set-theoretic operation

C1: Find an element’s attribute values
C2: Attribute distribution in a set / subset
C3: Compare attribute values between subsets
C4: Set memberships for specific attr. values
C5: Create a set of elements by attributes

The comparison matrix in Table 3 reveals how the techniques in the same category tend to have similar task support characteristics. As expected, this demonstrates the decisive influence of the visual encoding used by a technique on the types of tasks it supports. The matrix also reveals that certain techniques depend heavily on interaction in supporting their tasks. Clearly, there is no single technique that supports all the tasks. The choice of the technique to use for a specific problem requires extensive analysis of the problem domain and its data characteristics. This is important to determine the tasks that need to be supported and the actual scalability requirements.

6. Future Challenges and Opportunities

The techniques surveyed in Sect. 4 demonstrate the significant advances made in the past decade in visualizing sets and set-typed data. Nevertheless, research in this area is still in early stages, with many open problems and challenges that need to be addressed in the future. In the following we give some of these problems and provide a list of unexplored research directions that could help in addressing them.

6.1. Open Problems

Some of the issues we list are specific to certain techniques, while others are more generic in set visualization. Additionally, some problems are concerned with complicated forms of set-typed data.

Generating Euler diagrams with specific properties: There are no generic tools that indicate, for a given input, whether it is possible or not to generate diagrams that are well-matched, well-formed, area-proportional, and/or use certain shapes (e.g. circles or convex polygons). Rodgers [Rod13] elaborated on related open research questions in generating Euler diagrams. Tools that determine whether a diagram can be drawn with desired properties and propose alternative solutions to non-drawable cases (e.g. using shading or approximate areas) would improve the quality of the generated Euler diagrams and their applicability in various domains. In this regard, a high-level algorithm has been proposed to determine the drawability of a well-formed diagram and generates the diagram in that case [FFH08], but no implementation is available yet.

Scalability: As Table 3 shows, it is not always possible to support tasks if they have particular scalability requirements. Moreover, the scalability of certain techniques is severely limited, such as overlays. Improving upon these limits is necessary to address various real-world problems that involve a large number of sets and/or elements.

The role of ordering: By definition, set-typed data impose no inherent ordering neither on the elements nor the sets. However, the order in which sets and elements are depicted has a significant impact on the patterns and relations revealed by the visualization. Though reordering problems are usually NP-complete, a lot of work has been done for reordering generic matrices and node-link diagrams to reveal clusters and/or reduce clutter. This work need to be re-visited from sets perspective, e.g. by incorporating set-related data features such as element degrees. Also, more work is needed on the role of ordering in aggregation-based techniques.

Evaluation: There is a clear lack of empirical user studies that assess the effectiveness of different techniques in performing different tasks. Some comparative studies focus on techniques from the same category, such as overlays [AHRRC11, MHRS*13], while few studies compare techniques from different categories [CSR*14, RSA*14].

More evaluation work is needed to determine which techniques work well for which data characteristics and tasks, and to steer future research toward promising directions.

Visualizing sets in the context of other data types: Overlay techniques reveal set memberships of elements placed according to other data features. However, they offer limited possibilities as the layout of the overlays cannot influence the element placement. Designing set-aware visualizations can improve on this: As example, a set-aware graph layout would compute a node placement that reduces edge crossing and produces convex-shaped overlays at the same time. Further work is needed to visualize sets over elements in a timeline, a tree, or a multi-variate visualization.

Comparing multiple set families: In many scenarios, multiple instances of a set family are compared (e.g. how skill overlaps change across different companies). With few sets, small multiples of Euler diagrams help in comparing the set relations between the respective set families. As example, the comparison might involve finding which set relations change most / least across the different families. Dedicated techniques are needed to support such comparison tasks in a scalable way in the number of sets and families.

Time-varying set-typed data: As with many types of data, set-typed data can vary over time. For example, set memberships might change over time, leading to changes in set relations. Also, the attribute values of the elements might change over time even with static set memberships. Visualizing such changes is challenging, as the data is already complex. BubbleSets allow smooth re-computation of set overlays, making them suited to track the spatial distribution of set elements e.g. in an animated scatter plot. A technique similar to Parallel Sets was proposed to visualize object-group changes over multiple time steps [BvLA*11], however, allowing an element to belong to one set at a time.

Visualizing fuzzy and uncertain set memberships: Real-world data typically involve uncertainty that result in fuzzy set memberships. *disk diagram* [PP10] is a technique for analyzing fuzzy data using interactive visualization of fuzzy set operations. More work on both analytical and visual methods is needed to communicate the fuzziness in the data and study its effect on various set-related tasks.

6.2. Possible Opportunities

Here we list ideas and research directions that could improve on existing set visualization techniques.

Interaction opens new possibilities for addressing various challenges with analyzing and visualizing set-typed data. For example, when generating Euler diagrams, the user could specify certain constraints and properties or choose where to take a compromise when they are not satisfiable. Interactivity makes simplifying complex visualizations possible by showing certain information on demand and selecting certain parts to explore in more detail. It also facilitates various comparisons within one set family or across multiple families. Interaction allows influencing matrix reordering, e.g. to restrict changes to certain rows or columns. Finally, intuitive interactions allow sets to be combined using Boolean operations, performing multi-faceted search over a set of elements, or applying appropriate filtering and data reduction techniques to explore large set-typed data.

Coordinated multiple views can reduce the complexity of the data by showing information at multiple levels of detail. This can also provide complementary perspectives on the data (e.g. overlap matrix + spatial set distribution) to enrich the analysis.

Small multiples could provide solutions to visualizations that are severely limited in the number of sets, such as Euler diagrams, Mosaic Displays or Double-Decker plots. They can also be used to compare, for instance, data with certain attribute values to determine if they correlate with certain set relations or membership patterns.

Hybrid representations visualizations might be useful in certain cases, especially when the sets can be semantically divided in two groups. An example, in a 3 x 3 matrix or mosaic display of 3 sets, each cell or tile can additionally depict how its elements belong to another group of sets by using a different visualization such as a bar chart or an Euler diagram. Another example is combining matrix-based and frequency-based representations to visualize sets involved in an overlap and the overlap size.

Matrix-based representations are not fully exploited for visualizing set-typed data. They are relatively simple and clutter-free, and fairly scalable in the number of rows and columns. Moreover, there are several possibilities to encode multiple values in a matrix cell [ABHR*13]. This can be employed to show aggregated information on the elements and their attributes, as with aggregation-based techniques.

Analytical methods can transform large set-typed data into volumes suited for visualization and still preserving the most important information. In particular, several aggregations of the elements are possible based on their set-memberships, degrees, and attribute values. Similarly, intuitive set-operations can be used e.g. to aggregate multiple sets, or to replace a large family of sets with a smaller family over the same elements.

Identifying special cases and forms of set-typed data. As example, when the sets exhibit no intersection relations, treemaps would be a natural choice to visualize their containment hierarchy. Another example that arise in voting analysis, is when each element belongs to a constant number of sets, e.g. exactly to 3 sets out of 10. Such set memberships can be represented using three categorical variables which result in $\binom{10}{3} = 120$ non-redundant overlap combinations (many of them potentially empty). This is significantly lower than $2^{10} = 1024$ possible overlaps in the general case, and can be handled by categorical visualization techniques such as Parallel Sets.

Many other special cases can be identified in practical applications such as very sparse membership matrix, skewed or two-mode distribution of membership degrees, etc. The characteristics of these cases need to be studied extensively e.g. to identify if they satisfy certain Euler diagram drawability properties, can simplify existing visualization techniques, allow for new forms of visual representations or overlays, or lend themselves to new ways of aggregation.

7. Conclusion

The powerful and generic concepts of set theory make sets and set relations essential data models in several data analysis scenarios. Unlike common data types in information visualization such as graphs and trees, sets have been largely treated as data containers to group related elements or to illustrate overlaps between two or three groups. Nevertheless, a number of techniques have been devised to visualize sets and data related to them in the past decade. By emphasizing the notion of set-typed data, we have identified their specific characteristics as well as several measures and tasks commonly associated with this data type in visualization.

We have surveyed relevant literature on visualization techniques that can be applied to address these characteristics and tasks related to set-typed data, and have classified these techniques into seven categories, according to the main visual representation they use for depicting set relations. For each technique, we have analyzed which tasks it supports and its scalability with respect to the number of sets and elements. We have also outlined the general advantages and disadvantages of each representation, and which information they can represent from the data. This provides guidance for designers of set visualizations in choosing appropriate techniques for their data and tasks. Finally, we have examined major open problems in the area, and discussed various ideas that are worth investigating as opportunities to address open problems or to improve on state-of-the-art techniques. A visual browser of the surveyed techniques along with additional resources are available at http://www.setviz.net.

Acknowledgement This work was partially supported by the Austrian Federal Ministry of Economy, Family and Youth via CVAST, a Laura Bassi Centre of Excellence (No. 822746), and the Austrian Science Fund (FWF) via the KAVA-Time project (No. P25489).

References

[AAMH13] Alsallakh B., Aigner W., Miksch S., Hauser H.: Radial sets: Interactive visual analysis of large overlapping sets. *Visualization and Computer Graphics, IEEE Trans. on 19*, 12 (2013), 2496–2505. 1, 3, 10, 12, 13, 21

[ABHR*13] Alper B., Bach B., Henry Riche N., Isenberg T., Fekete J.-D.: Weighted graph comparison techniques for brain connectivity analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2013), ACM, pp. 483–492. 17

[AHRRC11] Alper B., Henry Riche N., Ramos G., Czerwinski M.: Design study of LineSets, a novel set visualization technique. *Visualization and Computer Graphics, IEEE Trans. on 17*, 12 (2011), 2259–2267. 8, 9, 16

[Bar69] Baron M. E.: A note on the historical development of logic diagrams: Leibniz, Euler and Venn. *The Mathematical Gazette 53*, 384 (1969), 113–125. 4

[BCH*13] Basole R. C., Clear T., Hu M., Mehrotra H., Stasko J.: Understanding interfirm relationships in business ecosystems with interactive visualization. *Visualization and Computer Graphics, IEEE Trans. on 19*, 12 (2013), 2526–2535. 13

[BCPS12] Brandes U., Cornelsen S., Pampel B., Sallaberry A.: Path-based supports for hypergraphs. *Journal of Discrete Algorithms 14* (2012), 248–261. 3

[BH11] Borgatti S. P., Halgin D. S.: Analyzing affiliation networks. *The SAGE handbook of social network analysis* (2011), 417–433. 13

[BR07] Benoy F., Rodgers P.: Evaluating the comprehension of Euler diagrams. In *International Conference on Information Visualization (IV)* (2007), pp. 771–780. 5

[Bra12] Brath R.: Multi-attribute glyphs on Venn and Euler diagrams to represent data and aid visual decoding. In *International Workshop on Euler Diagrams* (2012), pp. 122–129. 6

[BSR*13] Blake A., Stapleton G., Rodgers P., Cheek L., Howse J.: Improving user comprehension of Euler diagrams. *IEEE Symp. on Visual Languages and Human-Centric Computing (VL/HCC)* (2013), 189–190. 5

[BT06] Byelas H., Telea A.: Visualization of areas of interest in software architecture diagrams. In *ACM symposium on Software visualization (SOFTVIS)* (2006), ACM, pp. 105–114. 8, 9

[BT09] Byelas H., Telea A.: Visualizing metrics on areas of interest in software architecture diagrams. In *IEEE Pacific Visualization Symposium (PacificVis)* (2009), IEEE, pp. 33–40. 8

[BVKM*10] Buchin K., Van Kreveld M., Meijer H., Speckmann B., Verbeek K.: On planar supports for hypergraphs. In *Graph Drawing, LCNS vol. 5849* (2010), Springer, pp. 345–356. 3

[BvLA*11] Bremm S., von Landesberger T., Andrienko G., Andrienko N., Schreck T.: Interactive analysis of object group changes over time. In *International Workshop on Visual Analytics (EuroVA)* (2011), Eurographics, pp. 41–44. 16

[Can95] Cantor G.: Beiträge zur Begründung der transfiniten Mengenlehre. *Mathematische Annalen 46*, 4 (1895), 481–512. 2

[Cho07] Chow S. C.: *Generating and Drawing Area-Proportional Venn and Euler Diagrams*. PhD thesis, University of Victoria, Victoria, BC, Canada, 2007. 6

[Cla08] Clark J.: Twitter Venn. http://www.neoformix.com/2008/TwitterVenn.html, 2008. [Online; accessed Dec. 2013]. 5, 6

[CM84] Cleveland W. S., McGill R.: Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American Statistical Association 79*, 387 (1984), 531–554. 6

[CPC09] Collins C., Penn G., Carpendale S.: Bubble sets: Revealing set relations with isocontours over existing visualizations. *Visualization and Computer Graphics, IEEE Trans. on 15*, 6 (2009), 1009–1016. 1, 8, 9

[CR03] Chow S., Ruskey F.: Drawing area-proportional Venn and Euler diagrams. In *Graph Drawing* (2003), Springer, pp. 466–477. 5, 6

[CR05a] Chow S., Rodgers P.: Constructing area-proportional Venn and Euler diagrams with three circles. In *International Workshop on Euler Diagrams* (2005). 5, 6

[CR05b] Chow S., Ruskey F.: Towards a general solution to drawing area-proportional Euler diagrams. *Electronic Notes in Theoretical Computer Science 134* (2005), 3–18. 5, 6

[CSR*14] Chapman P., Stapleton G., Rodgers P., Micallef L., Blake A.: Visualizing sets: An empirical comparison of diagram types. In *International Conference on the Theory and Application of Diagrams (Diagrams)* (2014), Springer. In press. 5, 16

[CVW09] Collins C., Viegas F. B., Wattenberg M.: Parallel tag clouds to explore and analyze faceted text corpora. In *IEEE Symposium on Visual Analytics Science and Technology (VAST)* (2009), IEEE, pp. 91–98. 8, 9

[DHRRD12] Dörk M., Henry Riche N., Ramos G., Dumais S.: PivotPaths: Strolling through faceted information spaces. *Visualization and Computer Graphics, IEEE Trans. on 18*, 12 (2012), 2709–2718. 10, 11

[DvKSW12] Dinkla K., van Kreveld M., Speckmann B., Westenberg M.: Kelp diagrams: Point set membership visualization. *Computer Graphics Forum 31*, 3 (2012), 875–884. 4, 8, 9

[Eul68] Euler L.: *Lettres à une Princesse d'Allemagne sur divers sujets de physique et de philosophie, vol. 2, Lettres 102-108*. L'Académie Impériale des Sciences de Saint-Pétersbourg, St Petersburg, Russia, 1768. 4

[FFH08] Flower J., Fish A., Howse J.: Euler diagram generation. *Journal of Visual Languages and Computing 19*, 6 (2008), 675–694. 16

[FH02] Flower J., Howse J.: Generating Euler diagrams. In *Diagrammatic Representation and Inference (Diagrams), LNCS*, vol. 2317. Springer, 2002, pp. 285–285. 5

[FMH08] Freiler W., Matkovic K., Hauser H.: Interactive visual analysis of set-typed data. *Visualization and Computer Graphics, IEEE Trans. on 14*, 6 (2008), 1340–1347. 2, 12

[FRM03] Flower J., Rodgers P., Mutton P.: Layout metrics for Euler diagrams. In *International Conference Information Visualisation (IV)* (2003), pp. 272–280. 5

[Gre84] Greenacre M. J.: *Theory and applications of correspondence analysis*. Academic Press, 1984. 13

[Gur99] Gurr C. A.: Effective diagrammatic communication: Syntactic, semantic and pragmatic issues. *Journal of Visual Languages and Computing 10*, 4 (1999), 317–342. 4

[HB05] Heer J., Boyd D.: Vizster: Visualizing online social networks. In *IEEE Symposium on Information Visualization (INFOVIS)* (2005), IEEE, pp. 32–39. 8, 9

[HHH*89] Hamers L., Hemeryck Y., Herweyers G., et al.: Similarity measures in scientometric research: the Jaccard index versus Salton's cosine formula. *Information Processing & Management 25*, 3 (1989), 315–318. 3

[HKvK*13] Hurtado F., Korman M., van Kreveld M., Löffler M., Sacristán V., Silveira R. I., Speckmann B.: Colored spanning graphs for set visualization. In *Graph Drawing* (2013), Springer, pp. 280–291. 10

[Hof00] Hofmann H.: Exploring categorical data: interactive mosaic plots. *Metrika 51*, 1 (2000), 11–26. 12

[HRD10] Henry Riche N., Dwyer T.: Untangling Euler diagrams. *Visualization and Computer Graphics, IEEE Trans. on 16*, 6 (2010), 1090–1099. 7

[HST05] Howse J., Stapleton G., Taylor J.: Spider diagrams. *London Mathematical Society (LMS) Journal of Computation and Mathematics 8* (2005), 145–194. 6

[HSW00] Hofmann H., Siebes A. P., Wilhelm A. F.: Visualizing association rules with interactive mosaic plots. In *ACM SIGKDD international conference on knowledge discovery and data mining (KDD)* (2000), ACM, pp. 227–235. 12

[Huo08] Huo J.: KMVQL: a visual query interface based on karnaugh map. In *International Working Conference on Advanced Visual Interfaces (AVI)* (2008), ACM, pp. 243–250. 11

[IMMS09] Itoh T., Muelder C., Ma K.-L., Sese J.: A hybrid space-filling and force-directed layout method for visualizing multiple-category graphs. In *IEEE Pacific Visualization Symposium (PacificVis)* (2009), pp. 121–128. 9, 10

[KBH06] Kosara R., Bendix F., Hauser H.: Parallel sets: Interactive exploration and visual analysis of categorical data. *Visualization and Computer Graphics, IEEE Trans. on 12*, 4 (2006), 558–568. 12

[Kin] King G.: Find a proper pub. [Online; accessed Jan. 2014]. URL: http://www.findaproperpub.co.uk/. 9

[KLS07] Kim B., Lee B., Seo J.: Visualizing set concordance with permutation matrices and fan diagrams. *Interacting with computers 19*, 5-6 (2007), 630–643. 3, 7, 10, 11, 13

[KMK*08] Kestler H. A., Müller A., Kraus J. M., Buchholz M., Gress T. M., Liu H., Kane D. W., Zeeberg B. R., Weinstein J. N.: VennMaster: Area-proportional Euler diagrams for functional GO analysis of microarrays. *BMC Bioinformatics 9* (2008), 67. 5, 6

[Kof35] Koffka K.: *Principles of Gestalt Psychology*. Harcourt Brace, New York, NY, USA, 1935. 4

[Kos07] Kosara R.: Autism diagnosis accuracy - visualization redesign. http://eagereyes.org/criticism/autism-diagnosis-accuracy, 2007. [Online; accessed Dec. 2013]. 12

[KSB*09] Krzywinski M., Schein J., Birol İ., Connors J., Gascoyne R., Horsman D., Jones S. J., Marra M. A.: Circos: an information aesthetic for comparative genomics. *Genome research 19*, 9 (2009), 1639–1645. 3, 10

[KSJ*06] Koshman S., Spink A., Jansen B. J., Blakely C., Weber J.: Metasearch result visualization: an exploratory study. In *Canadian Association for Information Science Conference* (2006). 7

[KvKS09] Kaufmann M., van Kreveld M., Speckmann B.: Subdivision drawings of hypergraphs. In *Graph Drawing, LNCS vol. 5417* (2009), Springer, pp. 396–407. 3

[LLS05] Liu X., Luo M., Shneiderman B.: Visualization of sets. *Unpublished manuscript* (2005). 3, 11, 13

[LM13] Littlefield K., Monroe M.: Venn Diagram Plotter, Biological MS Data and Software Distribution Center. http://omics.pnl.gov/software/VennDiagramPlotter.php, 2013. [Online; accessed Dec. 2013]. 5, 6

[LRS10] Luboschik M., Radloff A., Schumann H.: A new weaving technique for handling overlapping regions. In *International Working Conference on Advanced Visual Interfaces (AVI)* (2010), ACM, pp. 25–32. 7

[Mäk90] Mäkinen E.: How to draw a hypergraph. *International Journal of Computer Mathematics 34*, 3-4 (1990), 177–185. 3

[MDF12] Micallef L., Dragicevic P., Fekete J.-D.: Assessing the effect of visualizations on bayesian reasoning through crowdsourcing. *Visualization and Computer Graphics, IEEE Trans. on 18*, 12 (2012), 2536–2545. 5, 6, 11, 21

[MHRS*13] Meulemans W., Henry Riche N., Speckmann B., Alper B., Dwyer T.: KelpFusion: a hybrid set visualization technique. *Visualization and Computer Graphics, IEEE Trans. on 19*, 11 (2013), 1846–1858. 8, 9, 16

[Mis06] Misue K.: Drawing bipartite graphs as anchored maps. In *Asia-Pacific Symposium on Information Visualisation (APVIS)* (2006), Australian Computer Society, Inc., pp. 169–177. 10, 11

[MM11] Mann K., Mann M.: In-depth analysis of the chicken egg white proteome using an LTQ Orbitrap Velos. *Proteome Science 9* (2011), 7. 4, 6

[MR09] Micallef L., Rodgers P.: Poster: Force-directed layout for Euler diagrams. *Compendium of IEEE Information Visualization (InfoVis)* (2009). http://www.eulerdiagrams.org/eulerForce. 5, 21

[MR14] Micallef L., Rodgers P.: eulerAPE: Drawing area-proportional 3-Venn diagrams using ellipses. *PLoS One* (2014). to appear, http://www.eulerdiagrams.org/eulerAPE. 5, 6, 21

[Pal92] Palmer S. E.: Common region: A new principle of perceptual grouping. *Cognitive Psychology 24*, 3 (1992), 436–447. 4

[Pod08] Podtelezhnikov A. A.: Proteins and DNA: Venn diagram of amino acid properties. https://sites.google.com/site/apodtele/aa_venn_diagram.jpg, 2008. [Online; accessed Jan. 2014]. 1

[PP10] Park Y., Park J.: Disk diagram: An interactive visualization technique of fuzzy set operations for the analysis of fuzzy data. *Information Visualization (IVS) 9*, 3 (2010), 220–232. 16

[RFS12] Rodgers P., Flower J., Stapleton G.: Introducing 3D Venn and Euler diagrams. In *International Workshop on Euler Diagrams* (2012), pp. 92–106. 6, 21

[RFSH10] Rodgers P., Flower J., Stapleton G., Howse J.: Drawing area-proportional Venn-3 diagrams with convex polygons. In *Diagrammatic Representation and Inference (Diagrams), LNCS*, vol. 6170. Springer, 2010, pp. 54–68. 5, 6, 21

[RGBS*11] Reid R. J., González-Barrera S., Sunjevaric I., Alvaro D., Ciccone S., Wagner M., Rothstein R.: Selective ploidy ablation, a high-throughput plasmid transfer protocol, identifies new genes affecting topoisomerase I-induced DNA damage. *Genome Research 21*, 3 (2011), 477–486. 4, 6

[RHSF14] Rodgers P., Howse J., Stapleton G., Flower J.: Drawing area-proportional Euler diagrams representing up to three sets. *Visualization and Computer Graphics, IEEE Trans. on 20*, 1 (2014). 5, 6, 21

[Rod13] Rodgers P.: A survey of Euler diagrams. *Journal of Visual Languages and Computing* (2013). 4, 16, 21

[RSA*14] Rodgers P., Stapleton G., Alsallakh B., Micallef L., Baker R., Thompson S.: A task-based evaluation of hybrid set and network visualizations. *Visualization and Computer Graphics, IEEE Trans. on* (2014), under review. 16

[RW97] Ruskey F., Weston M.: A survey of Venn diagrams. *Electronic Journal of Combinatoric 4* (1997), Dynamic Survey DS5 (revised in 2001 and 2005). 4

[RZF08] Rodgers P., Zhang L., Fish A.: General Euler diagram generation. In *Diagrammatic Representation and Inference (Diagrams), LNCS*, vol. 5223. Springer, 2008, pp. 13–27. 5, 21

[RZP12] Rodgers P., Zhang L., Purchase H.: Wellformedness properties in Euler diagrams: Which should be used? *Visualization and Computer Graphics, IEEE Trans. on 18*, 7 (2012), 1089–1100. 5, 21

[SA08] Simonetto P., Auber D.: Visualise undrawable Euler diagrams. In *International Conference Information Visualisation (IV)* (2008), IEEE, pp. 594–599. 7

[SAA09] Simonetto P., Auber D., Archambault D.: Fully automatic visualisation of overlapping sets. *Computer Graphics Forum 28*, 3 (2009), 967–974. 5

[SD08] Stapleton G., Delaney A.: Evaluating and generalizing constraint diagrams. *Journal of Visual Languages and Computing 19*, 4 (2008), 499–521. 6

[SDRP11] Stapleton G., Delaney A., Rodgers P., Plimmer B.: Recognising sketches of Euler diagrams augmented with graphs. In *International Workshop on Visual Languages and Computing (VLC)* (2011), vol. 17, pp. 182–196. 6

[SDS13] Sadana R., Dove A., Stasko J.: Poster: Whale sharks, Boolean set operations, and direct manipulation. In *Compendium of IEEE Information Visualization (InfoVis)* (2013). 11

[SFRH12] Stapleton G., Flower J., Rodgers P., Howse J.: Automatically drawing Euler diagrams with circles. *Journal of Visual Languages and Computing 23*, 3 (2012), 163–193. 5

[SGL08] Stasko J., Görg C., Liu Z.: Jigsaw: supporting investigative analysis through interactive visualization. *Information visualization (IVS) 7*, 2 (2008), 118–132. 10, 11

[Shn96] Shneiderman B.: The eyes have it: A task by data type taxonomy for information visualizations. In *IEEE Symposium on Visual Languages* (1996), IEEE, pp. 336–343. 4, 12

[SJUS08] Schulz H.-J., John M., Unger A., Schumann H.: Visual analysis of bipartite biological networks. In *EG Workshop on Visual Computing for Biomedicine* (2008). 10

[SOTM06] Shen Z., Ogawa M., Teoh S. T., Ma K.-L.: BiblioViz: a system for visualizing bibliography information. In *International Asia-Pacific Symposium on Visualization (APVIS)* (2006), pp. 93–102. 9, 10

[Spo93] Spoerri A.: InfoCrystal: A visual tool for information retrieval. In *IEEE Visualization* (1993), pp. 150–157. 13

[Spo04] Spoerri A.: MetaCrystal: visual interface for meta searching. In *ACM SIGCHI Conference on Human Factors in Computing Systems (CHI) - extended abstracts* (2004), vol. 24. no 29, ACM, pp. 1558–1558. 10, 13

[SRHZ11] Stapleton G., Rodgers P., Howse J., Zhang L.: Inductively generating Euler diagrams. *Visualization and Computer Graphics, IEEE Trans. on 17*, 1 (2011), 88–100. 5, 21

[Sta05] Stapleton G.: A survey of reasoning systems based on Euler diagrams. *Electronic Notes in Theoretical Computer Science 134* (2005), 127–151. 4, 6

[SWS*11] Steinberger M., Waldner M., Streit M., Lex A., Schmalstieg D.: Context-preserving visual links. *Visualization and Computer Graphics, IEEE Trans. on 17*, 12 (2011), 2249–2258. 8

[SZHR11] Stapleton G., Zhang L., Howse J., Rodgers P.: Drawing Euler diagrams with circles: The theory of piercings. *Visualization and Computer Graphics, IEEE Trans. on 17*, 7 (2011), 1020–1032. 5, 21

[TG80] Treisman A. M., Gelade G.: A feature-integration theory of attention. *Cognitive Psychology 12*, 1 (1980), 97–136. 6

[TS85] Treisman A., Souther J.: Search asymmetry: a diagnostic for preattentive processing of separable features. *Journal of Experimental Psychology: General 114*, 3 (1985), 285–310. 4

[Tve77] Tversky A.: Features of similarity. *Psychological review 84*, 4 (1977), 327. 3

[UJ12] Urbas M., Jamnik M.: Diabelli: A heterogeneous proof system. In *Diagrammatic Representation and Inference (Diagrams), LNCS*, vol. 7364. Springer, 2012, pp. 559–566. 6

[UJSF12] Urbas M., Jamnik M., Stapleton G., Flower J.: Speedith: a diagrammatic reasoner for spider diagrams. In *Diagrammatic Representation and Inference (Diagrams), LNCS*, vol. 7352. Springer, 2012, pp. 163–177. 6

[Ven80] Venn J.: On the diagrammatic and mechanical representation of propositions and reasonings. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 10*, 59 (1880), 1–18. 5

[VPF*14] Vihrovs J., Prūsis K., Freivalds K., Ručevskis P., Krebs V.: An inverse distance-based potential field function for overlapping point set visualization. In *International Conference on Information Visualization Theory and Applications (IVAPP)* (2014), SCITEPRESS, pp. 29–38. 8

[VRW13] Vehlow C., Reinhardt T., Weiskopf D.: Visualizing fuzzy overlapping communities in networks. *Visualization and Computer Graphics, IEEE Trans. on 19*, 12 (2013), 2486–2495. 10

[War12] Ware C.: *Information Visualization: Perception for Design, 3rd ed.* Elsevier, 2012. 4, 7

[Wik10] Wikimedia Commons: Official languages in Africa, 2010. [Online; accessed Jan. 2014]. URL: http://en.wikipedia.org/wiki/File:Official_languages_in_Africa.svg. 9

[Wil12] Wilkinson L.: Exact and approximate area-proportional circular Venn and Euler diagrams. *Visualization and Computer Graphics, IEEE Trans. on 18*, 2 (2012), 321–331. 5, 6

[Wit10] Wittenburg K.: Setting the bar for set-valued attributes. In *International Conference on Advanced Visual Interfaces (AVI)* (2010), ACM, pp. 253–256. 12

[WMLP12] Wittenburg K., Malizia A., Lupo L., Pekhteryev G.: Visualizing set-valued attributes in parallel with equal-height histograms. In *International Working Conference on Advanced Visual Interfaces (AVI)* (2012), ACM, pp. 632–635. 13

[WPS*11] Wang M., Plimmer B., Schmieder P., Stapleton G., Rodgers P., Delaney A.: SketchSet: Creating Euler diagrams using pen or mouse. *IEEE Symp. on Visual Languages and Human-Centric Computing (VL/HCC)* (2011), 75–82. 6

[WWC09] Wyatt D., Wynn D., Clarkson J.: Exploring spaces of system architectures using constraint-based classification and Euler diagrams. In *International Design Structure Matrix Conference (DSM)* (2009), pp. 141–144. URL: http://www-edc.eng.cam.ac.uk/tools/set_visualiser. 13

[XDC*13] Xu P., Du F., Cao N., Shi C., Zhou H., Qu H.: Visual analysis of set relations in a graph. *Computer Graphics Forum 32*, 3 (2013), 61–70. 9, 10

[ZKBS02] Ziegler E., Kunz C., Botsch V., Schneeberger J.: Visualizing and exploring large networked information spaces with Matrix Browser. In *International Conference Information Visualisation (IV)* (2002), IEEE, pp. 361–366. 11

Biography

Bilal Alsallakh is a research assistant at Vienna University of Technology, Austria, in the areas of information visualization and visual analytics. During his PhD, he developed the Radial Sets technique to visualize large overlapping sets [AAMH13]. His research interests encompass visual analysis of set-typed, relational, and categorical data, as well as software visualization, pattern recognition, and time-series analysis.

Luana Micallef is a Postdoctoral Researcher at the Helsinki Institute for Information Technology HIIT, Finland. She was previously a Research Fellow at the University of Kent, UK, where she completed her PhD on 'Visualizing Set Relations and Cardinalities Using Venn and Euler Diagrams'. She devised different diagram drawing software [MR09,MR14,MDF12] and presented her work at various institutes, including a featured talk for the Analytics and Big Data Society, Charlotte, NC. Her 2012 IEEE InfoVis paper [MDF12] on Euler diagrams for Bayesian reasoning was awarded an honourable mention. Luana co-edited the first special journal issue on visualization and reasoning using Euler Diagrams in 2014, served on the Programme Committee of the Euler Diagrams Workshop in 2014 and co-chaired the 2012 edition of the workshop.

Wolfgang Aigner is lecturer at St. Poelten University of Applied Sciences, Austria. His main research interests include visual analytics and information visualization. Wolfgang Aigner received his PhD degree and habilitation from the Vienna University of Technology in 2006 and 2013. He authored and co-authored several dozens of peer-reviewed articles as well as the recently published book "Visualization of Time-Oriented Data" (Springer, 2011) that is devoted to a systematic view on this topic.

Helwig Hauser is professor in visualization at the University of Bergen, Norway, where he is leading a research group on visualization. His research interests are diverse in visualization, including interactive visual analysis, the visualization of multi-dimensional and multi-variate data, and the application of visualization to the fields of medicine, geosciences, biology, fluid dynamics, etc. Helwig Hauser was the scientific director of the VRVis Research Center in Vienna, Austria, and an assistant professor at the Vienna University of Technology, Austria, from which he also received his graduate and doctoral degrees (in 1994 and 1998) as well as his habilitation (in 2003). He received several awards, including the bi-annual Heinz-Zemanek Award in computer science in 2006 and the Dirk Bartz Prize for visual computing in medicine in 2013. Helwig Hauser serves as paper chair of central conferences including EuroVis, InfoVis, and PacificVis, and is associate editor of major journals, including Computer Graphics Forum and IEEE TVCG. He is also member of several committees, including the EuroVis Steering committee.

Silvia Miksch is Associate University Professor and head of the Information and Knowledge Engineering research group at Vienna University of Technology, Austria. Her main research interests are Information Visualization and Visual Analytics, Process and Plan Management, Interaction Design, User-Centered Design, and Time. Silvia was professor and head of the Department of Information and Knowledge Engineering at Danube University Krems, Austria. In 2010 she established the awarded Laura Bassi Centre of Expertise CVAST. Silvia Miksch co-authored various state-of-the-art survey articles, and co-authored the recently published book "Visualization of Time-Oriented Data" (Springer, 2011) that is devoted to a systematic view on this topic. She served as paper co-chair of several conferences including IEEE VAST 2010 and 2011 and EuroVis 2012, and on the editorial board of several journals including Artificial Intelligence in Medicine (AIM-J, Elsevier), AI Communications (AICOM, IOS Press), and IEEE TVCG.

Peter Rodgers is a Reader in the School of Computing at the University of Kent, United Kingdom. His research interests are in diagrammatic information visualization. He has developed many set-based visualization tools, particularly for Venn and Euler diagrams. They include area-proportional methods [RFSH10, RHSF14], general embedding techniques and 3D diagrams [RFS12]. Peter has numerous publications on the topic [RZF08, SRHZ11, SZHR11, RZP12], including a recent survey paper on Euler diagrams [Rod13]. He is the recipient of a Royal Society Industrial Fellowship, and was principal investigator on the three year UK Research Council project 'Visualization with Euler Diagrams'. Peter was co-founder of the Euler Diagrams Workshop series in 2005, and General Chair of the Diagrams conference in 2012.