Subspace-Map: Interactive Visual Analysis for Subspace Data with a Map Metaphor

Jincheng Li, Chufan Lai, and Xiaoru Yuan

***Abstract*—Subspace analysis of high-dimensional data is extremely challenging due to the huge exploration space. We propose Subspace-Map, a novel approach with a map metaphor for interactive exploration of various subspaces. We utilize a subspace search algorithm to identify a moderate number of potentially valuable subspaces, each visualized as a city on the map. Similar cities are clustered into provinces and countries, highlighting common data and dimensional patterns that can guide users in constructing desired subspaces. With the map, users can grasp an overview of the exploration space and explore different subspaces via recommended tour routes in more detail. We demonstrate the effectiveness of Subspace-Map through cases with real-world data, experiments with user feedback, and a comparison with state-of-the-art subspace data visualizations.**

***Index Terms*—High-Dimensional Data, Subspace Analysis, Map Metaphor**

I. Introduction

In high-dimensional datasets, each data item is characterized by multiple attributes. However, not all attributes are equally informative. Redundant attributes can obscure crucial patterns, like clusters and correlations. For instance, in analyzing the relationship between education level and income, attributes like weight and height are usually less insightful. The more time we spend studying redundant attributes, the less likely it is to reveal the information of interest. Hence, experienced analysts often start by selecting a small, task-relevant subset of dimensions. The data space created by a subset of dimensions is known as a **subspace**.

Subspaces are of two types: axis-aligned and non-axis-aligned. Axis-aligned subspaces have axes parallel to the original dimensions, while non-axis-aligned subspaces use axes representing weighted combinations of these dimensions. Non-axis-aligned subspaces, often produced by linear dimensionality reduction methods, are adept at uncovering patterns like clusters but typically do not maintain dimensional semantics. This paper focuses exclusively on axis-aligned subspaces for their user-friendliness and interpretability. We refer to them as "subspaces" for short.

Subspace analysis presents significant challenges, including:

- **Overwhelming exploration space.** In combinatorial terms, this issue manifests as the combinatorial explosion problem. For a d-dimensional dataset, there are altogether $2^d - 1$ subspaces, each of which may present unique data patterns. The exploration workload doubles with each added dimension, making it easy for users to be overwhelmed without a clear mental map.
- **Complex dimension-data pattern interplay.** Adding or removing a single dimension seems like a subtle alteration, but it can drastically alter the data structure. Analysts, unable to predict these changes, might inadvertently disrupt important data patterns.
- **Lack of planning in exploration.** After investigating one subspace, deciding the next one to explore is challenging. Analysts may attempt to locate a similar candidate that minimizes changes in dimensions and data. However, the unpredictability of data structures within subspaces often leads to a trial-and-error approach.

Various algorithms have been developed to identify subspaces with valuable data clusters [1]–[3]. However, they often yield redundant results, necessitating further organization and analysis with the assistance of visualization [4], [5]. These algorithms also lack guidance for dimension selection, complicating user adjustments. While there are visual analytic methods to assist in subspace exploration, they have limitations: some require inefficient manual planning [6], [7], while others only work with 2D subspaces [8], restricting their effectiveness and applicability.

We set three goals to address the challenges mentioned:

- **G1:** Help users build up mental maps of the subspace exploration space.
- **G2:** Reveal the interplay between dimension and data patterns to guide dimensional decisions.
- **G3:** Aid users in scheduling their exploration with a series of smoothly transitioned subspaces.

We propose Subspace-Map, a visualization approach employing map metaphors to provide an overview of subspaces and guide users in the exploration. It visualizes the exploration space as a geographic map, representing each subspace as a city (**G1**). The landscape of each city represents the data patterns shaped by specific natural factors (dimension combination). By comparing landscapes, similar cities are clustered into provinces and countries. We extract common dimensions and data patterns in each cluster, revealing the inherent interplay between dimensions and data patterns (**G2**). We also construct tour routes, allowing users to schedule their subspace "trips" to explore different landscapes (**G3**).

Jincheng Li and Xiaoru Yuan are with Key Laboratory of Machine Perception (Ministry of Education), School of Intelligence Science and Technology, Peking University. E-mail: {jincheng.li, xiaoru.yuan}@pku.edu.cn.

Xiaoru Yuan is also with National Engineering Laboratory for Big Data Analysis and Application, Peking University.

Chufan Lai is with Technology and Engineering Center for Space Utilization, Chinese Academy of Science. E-mail: laichufan@csu.ac.cn.

Jincheng Li and Chufan Lai contribute equally to this work.

Xiaoru Yuan is the corresponding author.

Manuscript received April 19, 2021; revised August 16, 2021.

The remainder of the paper is structured as follows. Section II provides a literature review on high-dimensional data visualization, subspace analysis, and map metaphor visualizations. Section III discusses design considerations and introduces Subspace-Map's conceptual design. Details regarding map construction are presented in Section IV. We describe our prototype system's user interface and interactions in Section V. Section VI evaluates Subspace-Map through two case studies, a user study, and a comparison with state-of-the-art approaches. Section VII discusses current limitations and potential improvements. The final section concludes the paper.

II. Related Work

In this section, we first describe high-dimensional data visualizations that identify key dimensions. We then introduce subspace mining techniques and associated visualization methods. Finally, we review visualizations using map metaphors.

A. Dimension Selection in High-Dimensional Data

Dimension selection, critical for reducing dimensionality while retaining important features, demands extensive data familiarity, a condition often lacking in data analysis. Research addressing this challenge falls into three broad categories.

The first category encompasses methods grounded in dimension similarity. In the early research of parallel coordinates [9], Yang et al. [10] suggested clustering similar dimensions and using the centroid or average of each cluster as a representative dimension. Turkay et al. [11] applied dimensionality reduction and statistical modeling to create representative factors. Zhang et al. [12] utilized correlation strengths and established more refined rules for dimension clustering.

The second category focuses on selecting crucial dimensions based on quality metrics. The Rank-by-Feature Framework [13] enables users to rank dimensions using statistical criteria, useful in scatterplot matrices (SPLOM) [14] and parallel coordinates. Scatterplot pattern salience measures [15], including Scagnostics [16], rank plots in SPLOMs [8] to enhance exploration efficiency. Sedlmair et al. [17] proposed a taxonomy for visual cluster separation in scatterplots, guiding the design and evaluation of cluster separation measures. In parallel coordinates, metrics have been introduced to detect the inter-axis patterns [18] and rank ordering schemes [19]. PC-Expo [20] identifies twelve common analysis tasks, offering an interactive framework for axes reordering and local pattern detection. We refer to [21], [22] for a comprehensive review. With machine learning advancements, more advanced methods have emerged. Drawing inspiration from contrastive principal component analysis (cPCA) [23], ccPCA (contrasting clusters in PCA) [24] calculates dimension contributions to each cluster. Similarly, Knittel et al. [25] introduced a neural network-based model for extracting dimensions correlated with item groups. Our method falls into this category, extracting crucial dimensions by evaluating their dominance in patterns.

In contrast to automated methods, the third category emphasizes interactive dimension selection without preset notions of interest. Voyager [26] and its advanced version [27] offer free selection from a complete list, also suggesting potentially overlooked dimensions. Sarvghad et al. [28] achieved similar objectives by displaying explored dimension coverage. Turkay et al. [29] introduced dimension brushing for dual-space analysis, which Yuan et al. [7] later expanded into a hierarchical exploration framework. Cheng and Mueller [30] presented a unified layout for both data items and dimensions, enabling users to simultaneously observe data item patterns, dimension patterns, and their interrelationships.

B. Subspace Mining and Visualization

Dimension selection methods effectively identify a limited number of subspaces or those with fewer dimensions. However, for high-dimensional subspaces, subspace mining techniques become essential. These techniques aim to discover subspaces with interesting patterns, often hidden clusters. For example, CLIQUE [31], a pioneering approach, uses a combination of density- and grid-based clustering with an apriori-style technique for identifying clusterable subspaces. RIS [32] ranks subspaces using a quality criterion based on the density-based clustering concept of DBSCAN [33]. These are generally known as subspace clustering algorithms. Please refer to [1]–[3] for a systematic review.

Subspace clustering often results in a high volume of redundant outcomes. Hund et al. [34] visually assessed results focusing on non-redundancy, object and dimension coverage, and clustering characteristics. Tatu et al. [4] developed a visual analytics system to organize redundant subspace candidates, illustrating their relationships through dimension and data similarities. TripAdvisorND [35] showcases a projection highlighting dimension differences, while Pattern Trails [5] adopts a 1D layout to track data changes across subspaces. A comprehensive comparison is presented in Section VI.

C. Map Metaphor for Non-Spatial Data Visualization

Maps, essential for depicting spatial relationships, vary in types like choropleth and road maps, each conveying different information. Their familiarity and ease of understanding make them suitable for representing non-spatial information. Spatialization refers to creating graphic representations for a high-dimensional information space and transforming the information into its essential components [36]. It supports "the viewer's intrinsic comfort with everyday concepts of human spatial orientation and wayfinding to guide the exploration and interpretation of the representation" [37]. Skupin and Fabrikant have comprehensively reviewed spatialization methods for creating non-geographic visualizations [38], [39].

GMap [40] a trailblazer in this field, was initially developed for community identification in networks and later applied to dynamic graph analysis [41] and video content study [42]. Map metaphors are widely used in set visualization and social media data analytics. MetroSets [43] employs the metro map metaphor for set systems visualization, depicting common elements as metro interchanges. MosaicSets [44] creates Euler-like diagrams for set systems using hexagonal or square grids. In social media analysis, Chen et al. [45] introduced D-Map for analyzing user-centric information diffusion patterns, while Chen et al. [46] proposed R-Map for studying information

reposting processes. A recent survey [47] provides a comprehensive overview of map-like visualizations. Differing from existing approaches, Subspace-Map supports multi-level exploration with automatically extracted and organized patterns at various granularities. It also introduces a transportation concept, allowing users to follow pattern changes across subspaces in a recommended or user-defined order.

III. The Design of Subspace-Map

In this section, we outline our design considerations and justify the use of a map-like representation. Then, we detail the visual encodings and explain how the design aligns with our initial considerations.

A. Design Considerations

We formulate three goals in Section I. To achieve these, our design needs to meet various requirements, some of which align with those identified in previous subspace visual analysis research [4], [5].

DC1: Limited subspace overview with relationships. Due to the combinatorial explosion in high-dimensional data, displaying all subspaces within the same view is nearly impractical. An effective mental map (**G1**) requires showing the relationships between a limited number of subspaces.

DC2: Selecting potentially valuable subspaces. To meet DC1, algorithms are needed to limit the number of visualized subspaces. These subspaces should be informative and representative in the exploration space.

DC3: Summarization for shared features. Summarization alleviates scalability issues by enabling higher-level analysis and reveals the interplay between data and dimensions (**G2**), crucial for understanding subspace relationships.

DC4: Guided exploration recommendations. Recommending a few representative subspaces helps users quickly comprehend different subspaces. Suggesting a tour path with gradual changes aids in preventing confusion from abrupt data changes. These recommendations support users in identifying and sequencing their exploration of target subspaces (**G3**).

B. Design Choices and Decisions

We refined our design through a case study with the 8D Forest Fires data, including 247 subspaces (details in Section VI-A). Initially, we employed a straightforward method: a 2D projection with each point representing a subspace [4]. However, this approach revealed two major issues.

First, there is a conflict between displaying trends and revealing details. To expose trends, the overview should efficiently accommodate multiple subspaces (**DC2**). It also needs to deliver detailed information in the overview to help users identify areas of interest for further exploration. While projections can display numerous subspaces, they risk becoming cluttered, making it hard to allocate sufficient space for each subspace's details. This issue affects both 2D [4] and 1D [5] projections. The core problem is using screen distance to indicate similarity without accounting for the visual space needed by each subspace. Overlaps are common when a subspace requires more space than a dot. Compared to subtle changes in screen distances, users tend to focus more on significant features like clusters and outliers. Therefore, reducing occlusions by adjusting screen distances can be effective, even if it slightly compromises the accuracy of subspace similarity representation.

Second, non-expert users may find subspace concepts difficult to understand and manipulate. Despite having needs for subspace exploration, such as selecting data dimensions, they are often unfamiliar with these concepts. A subspace, defined by its dimension combination, complicates matters further when users compare various combinations or consider higher-level concepts like a set of subspaces. This situation calls for adopting visual metaphors that simplify subspace analysis.

We identify two key requirements for the overview's design. First, it must be space-efficient and occlusion-free, capable of displaying hundreds to thousands of distinct subspaces. Second, it should use user-friendly metaphors to enhance understanding and communication.

1) Visual Style of the Overview: Geographic maps, widely recognized and capable of hosting large, occlusion-free subunits, offer a scalable hierarchy (e.g., states and cities) [48] suitable for complex information spaces. People, familiar with maps from an early age, can effortlessly interpret them. Hence, map metaphors are frequently used to visualize diverse information spaces like the World Wide Web [49], scientific literature [40], and social media [45]. Studies [50] indicate that geographic metaphors aid in understanding non-spatial information, a process known as **spatialization** [38], [39]. To address issues in 2D projections, we opted for a map-style overview, a novel approach in subspace visualization.

Fabrikant and Buttenfield's cognitive framework [37] suggests that spatialization requires multiple levels of distinguishable concepts. However, subspaces lack a natural hierarchy and often feature subtle differences despite their large numbers, posing a challenge to spatialization. While filtering (**DC2**) and summarization (**DC3**) ease the challenges, a space-efficient map style remains essential.

2) Map-Like Visualizations: Map-like visualizations can be categorized into four types based on visual primitives [47]. Point-based imitations draw on map symbols such as location labels and icons but are prone to visual occlusions. Line-based versions represent data categories and connections akin to geographic borders and road networks [51]. Field-based imitations construct "terrains" with isolines to show data trends [52]). However, categories, connections, and continuity are not inherent to subspace data. Area imitations, mirroring administrative divisions, suit our needs to display individual subspaces and higher-level summaries (**DC3**). Like geographic regions, these data areas exhibit relationships through distances and do not overlap (**DC1**).

There are three types of area imitation maps: geometric hulls, geometric tessellation, and regular grids [47]. Geometric hulls group points without accommodating individual data items, making them unsuitable for our needs. Geometric tessellation (e.g., Voronoi diagrams) creates irregular cells for each data point. In contrast, grid-based techniques align data points on a regular grid. We chose regular grids to ensure

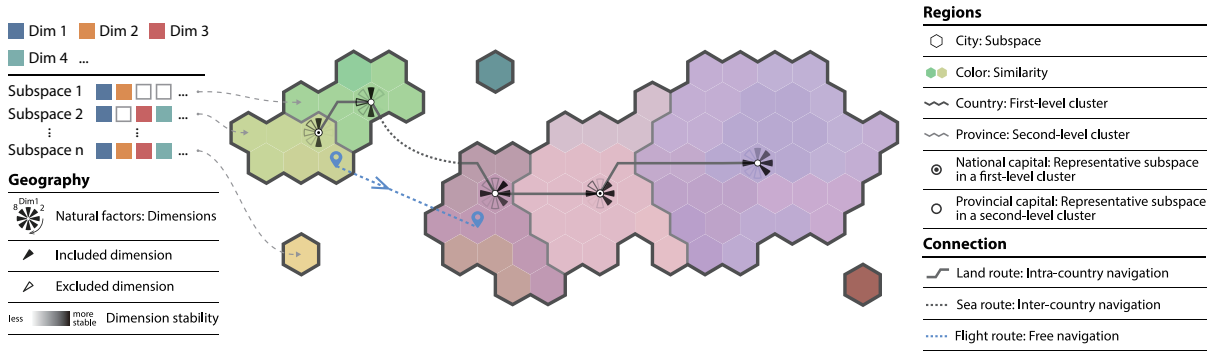


Fig. 1. Visual encodings of Subspace-Map. Each hexagon symbolizes a subspace, and each region signifies a cluster of subspaces. Various map metaphors, such as administrative divisions and transportation, are employed for enhanced informativeness and functionality.

uniformity among subspaces, avoiding the inequality implied by varying cell shapes and sizes. Regular grids also simplify standardizing the display for each subspace.

Of the regular grid types—triangular, square, and hexagonal [53], [54]—research [47], [55] suggests hexagonal grids are superior due to more adjacency relationships (each hexagon has six neighbors) and fewer cartographic errors. Thus, we use regular hexagonal grids for spatializing subspace data. While not as precise as projections in showing distances, they effectively represent relative data relationships with proximal and distant cells (**DC1**).

C. The Map Metaphors

Before delving into the algorithms, we explain how geographical terms correspond to subspace concepts in Subspace-Map. In Subspace-Map, the exploration space is an unknown **world** comprising **lands** (subspaces) and **oceans** (distances between subspaces). As visualization designers, we function as **cartographers**, **developers**, and **tour guides**.

1) Geography: As cartographers, we measure the topography and create a map. Given that it is a fictional world, there is no ground truth about how the subspaces should be arranged. Leveraging Gestalt Principles [56], we place similar subspaces close together to indicate similarity (**DC1**, Fig. 1).

Each subspace, envisioned as land, has a **landscape** defined by its data distribution, the primary interest of **tourists** (users). Dimension settings, analogous to **natural factors**, shape these landscapes. Apart from sightseeing, tourists often compare landscapes to understand what kind of natural factors are responsible for local features (**DC3**). Lands with similar landscapes form a **continent** (a cluster of subspaces), like dry and freezing Greenland or humid and hot Amazon Rainforest. **Islands** (outliers) are lands with highly distinctive landscapes. They are set apart from continents by **oceans**, representing void spaces without any specific meaning.

2) Regions: As developers, we establish cities and set up administrative divisions. Lands with appealing scenery are ideal candidates for developing tourist **cities**, corresponding to subspaces with valuable data patterns. These cities are chosen based on an aesthetic standard (subspace interestingness metric) likely reflecting public preference (**DC2**). Only these cities are displayed on the map.

As it is impractical and unnecessary for tourists to visit every city, we introduce higher-level divisions like **provinces** and **countries** to highlight regional characteristics (**DC3**). These divisions represent clusters at different granularities. A **capital city** (representative subspace) stands for its region in each, while unique **municipalities** signify outliers in a cluster.

This approach results in a three-tiered hierarchy spanning national, provincial, and urban levels. High-level summarizations reveal inter-subspace relationships; low-level details show data and dimension patterns.

3) Transportation: As tour guides, we organize sightseeing paths for tourists (**DC4**). We set up **flight routes** (the blue path in Fig. 1) for long-distance travel between different countries. These routes allow quick travel from one city to another, like flying directly from Los Angeles to London. However, this swift travel can lead to a metaphorical "jet lag" due to abrupt environmental changes. In our context, this represents the confusion from sudden visual changes when switching between very different subspaces. As a result, while flight routes facilitate quick journeys, they can make it hard for tourists to trace abrupt changes in data patterns and further reason based on the dimension changes.

As an alternative, we also establish **land routes** and **sea routes** (the black path in Fig. 1), connecting neighboring cities on the same continent and across seas, respectively. These routes involve traveling through all intermediate cities and are slower. Since neighboring cities usually have similar landscapes, they offer a more gradual and comprehensible tour, making understanding the differences between the origin and destination easier.

IV. The Construction of Subspace-Map

This section details the algorithms used for processing data and constructing Subspace-Map. As depicted in Fig. 2, we first identify potentially valuable subspaces. Afterward, we compute similarities between these subspaces, cluster them based on these similarities, and extract features from each cluster. Utilizing the clusters and similarities, we develop a map layout algorithm to appropriately position each subspace within a hexagonal grid.

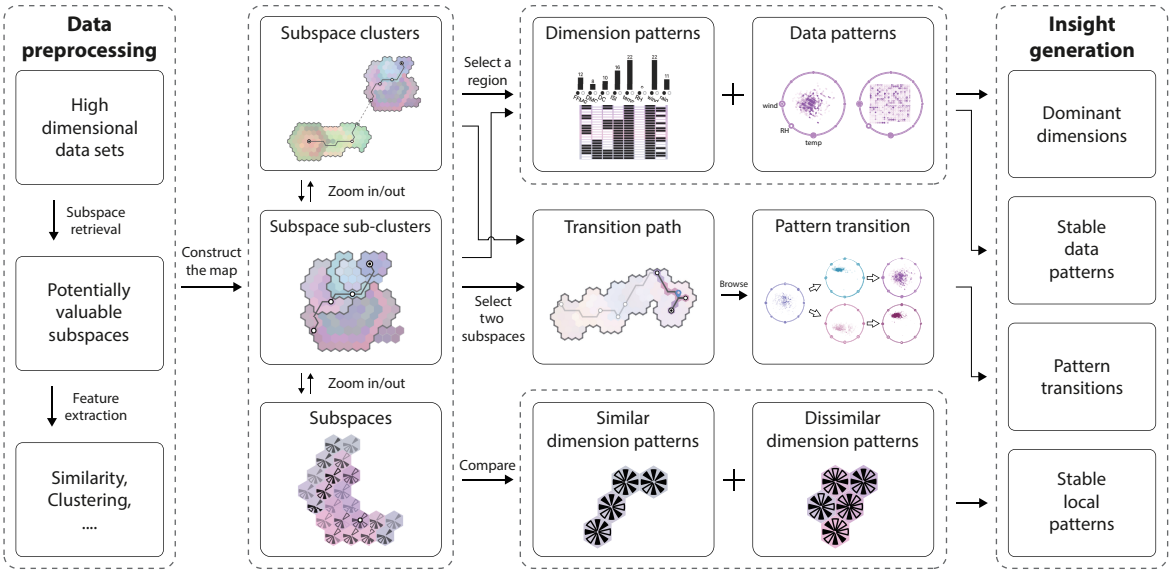


Fig. 2. Subspace-Map workflow. After constructing the map based on the input data, an overview showing the clusters of subspaces is provided. Users can explore hierarchically at the cluster, sub-cluster, and subspace levels. By analyzing various kinds of patterns and pattern transitions, they can gain insight into the dominant dimensions, stable data patterns, etc.

A. Data Preprocessing

In data preprocessing, the first step is normalization, essential for enabling cross-dimensional comparison. We then identify a limited number of potentially valuable subspaces and measure their similarities (**DC1**). Using these similarities, we cluster the subspaces. Finally, we extract features from each cluster to uncover common patterns (**DC3**).

Subspace retrieval. We leverage a subspace clustering algorithm to retrieve valuable subspaces (**DC2**). Selecting one from the plethora available is challenging. We set criteria based on Jahirabadkar and Kulkarni's categorization scheme [57], considering cluster orientation, overlap of dimensions or objects, search methods, and use of grid.

We focus on axis-aligned subspaces for user-friendliness and interpretability, favoring algorithms with axis-parallel cluster orientations. While keeping the number of subspaces manageable, we should not overlook potentially valuable subspaces. This means allowing dimensions and objects to overlap, enabling the algorithm to identify significant patterns in all subspaces. Regarding search methods, we choose the bottom-up approach over the top-down, as the latter assigns each item to only one cluster. Finally, we prefer density-based methods over grid-based ones for grid usage, as they offer more flexibility and do not restrict cluster shape and size.

In summary, our ideal algorithm should be axis-parallel, permit overlaps in dimensions and objects, and use a bottom-up, density-based approach. To further refine our choice, we exclude density-based algorithms dependent on a global density threshold, like SUBCLU [58], to prevent missing subspaces due to varying dimensionalities.

We chose the SURFING (SUbspace Relevant For clusterING) [59] algorithm. SURFING identifies subspaces with non-uniform distance distributions based on each data item's k-th nearest neighbor distance, suggesting meaningful data structures like clusters. It is unbiased in dimensionality and accommodates various cluster structures. Its use in previous subspace visual analysis research [4], [5] allows for visual comparability with those works. However, our method is not exclusively tied to SURFING; any algorithm meeting our criteria is suitable.

On the other hand, subspace clustering algorithms like SURFING are mainly intended for quantitative data. Categorical data, lacking a natural sense of distance or density, complicate subspace clustering. Converting categorical data into quantitative form using techniques like one-hot encoding is a viable solution to handle mixed or purely categorical data.

Similarity measurement. Traditional similarity measures like Euclidean distance are ineffective for comparing subspaces due to varying data distances across different dimensionalities. Jäckle et al. [5]'s projected distance approach is inaccurate due to inevitable information loss. Following Tatu et al. [4], we measure similarity by comparing data topology, i.e., the k-NN (k-Nearest Neighbors) relationship of data items in subspaces (**DC1**).

Specifically, we generate a k-NN list for each data item within a subspace. We regard subspace similarity for each item as the percentage of agreement (Jaccard Distance) between its two k-NN lists. The overall similarity is the average of these item-wise similarities. From this, we derive a j-NN list for each subspace, setting j to 6 to match the hexagonal grid's neighbor count. This helps identify common patterns in subspaces on a smaller scale (**DC3**).

Clustering. To achieve summarization at different granularity levels, we hierarchically cluster the subspaces using DBSCAN [33], a density-based method adept at identifying clusters of various structures (**DC3**). We implement two clustering levels corresponding to the map's conceptual divisions of countries and provinces. Using the sorted j-dist graph [33], which ranks each subspace by its distance to the j-th nearest neighbor, we perform top-level clustering to define countries

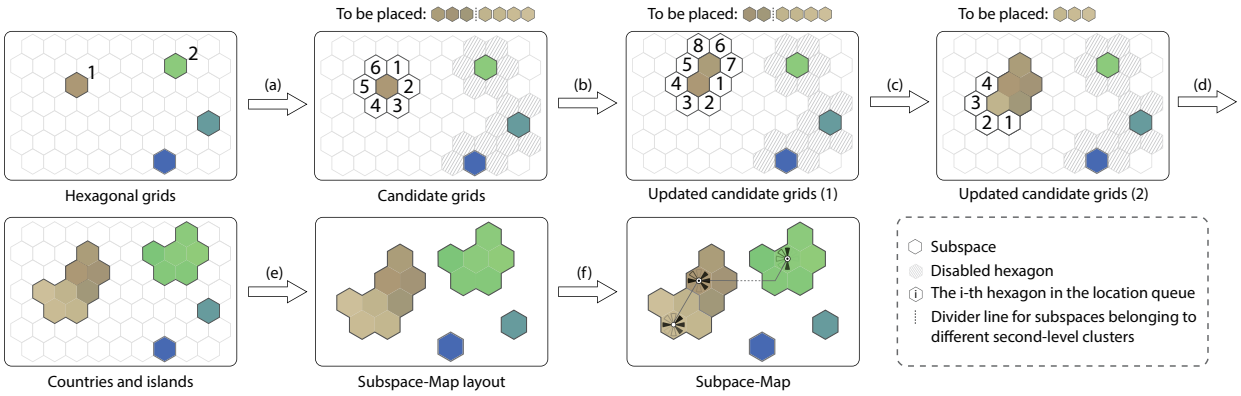


Fig. 3. The construction process of Subspace-Map: (a) initialize the location queue; (b) update the queue after each placement; (c) reset the queue for a new cluster; (d) continuously tiling; (e) reduce redundant map space; and (f) render map metaphors.

and islands. Then, we apply a second clustering level within each country to delineate provinces and municipalities.

Data pattern. We assess item-wise subspace similarity by comparing two k-NN lists for the same data item. Items with high neighborhood agreement are deemed stable across both subspaces. From this, we identify data items with consistent k-NN lists across all subspaces within a cluster. This yields the cluster's **data pattern**, highlighting the common data structure shared by its member subspaces (**DC3**).

Dimension pattern. Like data patterns, dimension patterns show the common dimensions shared among subspaces in a cluster. We calculate the occurrence frequency of each dimension within the cluster's subspaces. Since a subspace's data pattern is influenced by its dimensions, and subspaces in the same cluster have similar data patterns, the dimensions commonly included or excluded are crucial for these similarities. We identify these as **featured dimensions** by applying high and low thresholds for dimension occurrences.

Representative subspaces. A representative subspace is required to display each cluster's data and dimension patterns. This is chosen as the one at the cluster's center, having the highest average similarity with other members. We conceptualize this representative subspace as the capital city.

B. Map Construction

While numerous map layout algorithms exist [47], none fully suit our context (Section III-B). Hence, we developed a new layout algorithm for Subspace-Map. It involves two steps: first, positioning an anchor point for each country and island on the map, determining the relative positions of top-level clusters and outliers. Second, we arrange cluster members according to a predefined similarity-based traversal order.

1) Anchor Placement and Region Traversal: Initially, we create a dimension-reduced projection of all subspaces for the strategic placement of countries and islands (V in Algorithm 1). We use each country's capital as its anchor point and estimate its position on the hexagonal grid. We use the global dimensionality reduction technique MDS [60] over local techniques like t-SNE [61], as MDS better preserves all-point-pair distances for a more accurate projection [62]. Moreover, Ingram et al.'s work shows MDS's effectiveness with large data structures [63]. Based on anchor point locations and the number of subspaces, we calculate the map's size, i.e., its grid's rows and columns, ensuring adaptability.

Next, we set the traversal order for different regions (T in Algorithm 1). At the top level, we sequence national capitals and islands. For the second level, we arrange cities within each country. We start with an empty list and sequentially add the city closest on average to those already listed, ensuring similar

Algorithm 1 Map Layout Algorithm

Input:
A hexagonal map Map;
A list of anchor points $V[i]$ with initialized locations $V[i].loc, i = 1, 2...N_a$, N_a is the number of anchor points;
A list of traversal order lists for countries and islands $T[i], i = 1, 2...N_a$;
A list of city-province objects for countries and islands $O[i], i = 1, 2...N_a$;
Output:
A list of hexagonal grid cells that lists $V[i]^{'}$ for countries and islands, with assigned locations $V[i]^{'}.loc, i = 1, 2...N_a$;

for $i = 0; i < N_a; i++$ **do**
  $V^{'}[i]$.push($V[i]$)
end for
for $i = 0; i < N_a; i++$ **do**
  if $T[i].length == 1$ **then** ▷ This is an island
    continue
  end if
  // Step 1: Calculate the disabled cell list
  $G_{disabled}$ = []
  for $j = 0; j < N_a \&\& j != i; j++$ **do**
    Calculate the adjacent cells $G_{adjacent}$ of $V^{'}[j]$
    $G_{disabled}$.concat($G_{adjacent}$)
  end for
  // Step 2: Maintain the location queue Q
  for $j = 1; j < T[i].length; j++$ **do**
    for each adjacent grid cell g of $T[i][j-1]$ **do**
      if g is not in $G_{disabled}$ && g is empty **then**
        Q.enqueue(g)
      end if
      $V^{'}[i]$.push(Q.dequeue())
      if $j \neq T[i].length - 1 \&\& O[i][j] != O[i][j+1]$ **then** ▷ Two cities belong to different provinces
        Q.clear()
      end if
    end for
  end for
end for
// Step 3: Make the map compact
Remove empty rows/columns not causing bordering
Center and enlarge Map

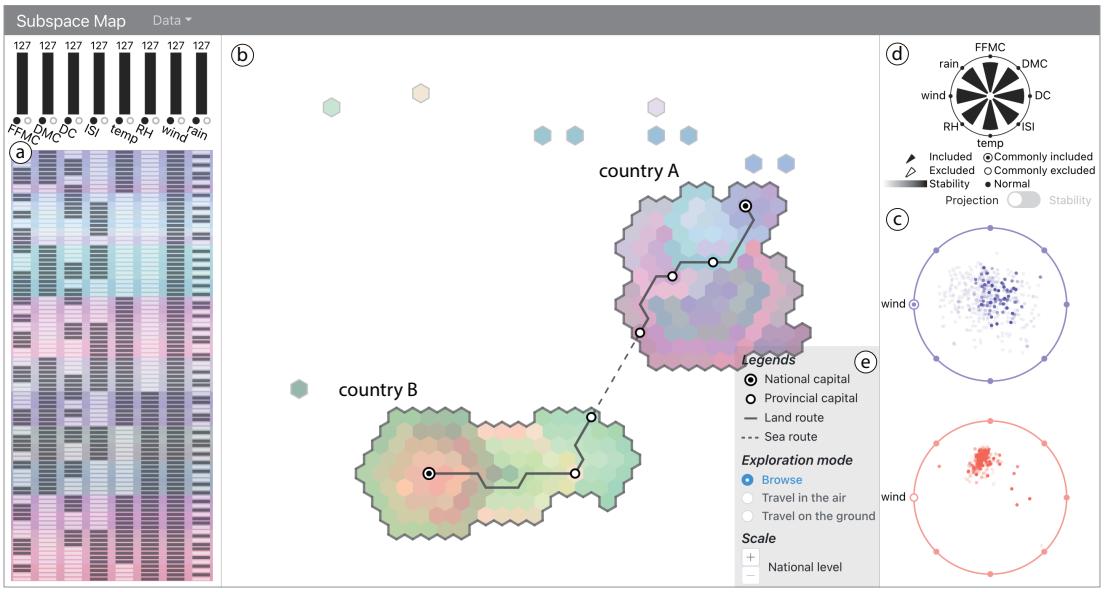


Fig. 4. The user interface of Subspace-Map. (a) Subspace List View shows each subspace's dimensions, with black/white indicating presence/absence of dimensions. (b) Map View displays subspace distribution based on similarities. (c) Map Detail View presents dimension and data patterns for a selected region. (d) and (e) explain visual encodings and map metaphors, and let users switch exploration modes.

cities are positioned near each other. Since the national capital, representing the cluster, has the highest average similarity to other members, it is always first on the list. We repeat this process until all cities in a country are included, treating each country independently. As islands contain only one city, they do not require traversal.

2) Grid Tiling: After setting the anchor points, we position cities within the same country based on their traversal order. We use a location queue to track available grid cells (Fig. 3a, Q in Algorithm 1), operating on a first-in, first-out (FIFO) basis. We allocate the first cell in the queue for each subspace, updating the queue continuously. Upon using a cell, we add its unoccupied, enabled adjacent cells to the queue clockwise. Disabled cells are used to prevent adjacent placements of different countries and islands (Algorithm 1-Step 1). For each country and island, except the one being constructed, we mark its adjacent cells as disabled to ensure no new subspaces are placed next to them.

The location queue starts with the second subspace in the traversal order, as the anchor point (the first subspace) is already positioned by the projection (Algorithm 1-Step 2). For each incoming subspace, we add the available neighboring cells of the previous subspace (not disabled or occupied) to the queue. The first cell in the queue is then assigned to the new subspace. Next, we determine if the upcoming subspace is in the same second-level cluster as the current one. If so, the queue updates as usual (Fig. 3b). If not, we clear the queue and start anew with the current subspace (Fig. 3c). This ensures tight packing within the same cluster. We avoid starting new queues for second-level outliers (municipalities), as they are usually placed at the end of the list, and individual queues could lead to an elongated tail.

After constructing all countries and islands (Fig. 3d), we compact the layout by reducing the gaps (Fig. 3e) and then center and enlarge the remaining grids to fully utilize the view space (Algorithm 1-Step 3). Since compaction might alter spatial relationships, we use colors to represent subspace similarity, offsetting potential information loss. We project the subspaces into 3D using the distance matrix, with each axis corresponding to an RGB color parameter. This assigns each subspace a unique color based on its projected coordinates.

3) Map Enriching: After generating the map layout, we enhance it with map metaphors like capital cities, natural factors, and travel routes for greater informativeness (Fig. 3f).

Each cluster's representative subspace is depicted as a capital city. The dimension combination/pattern and data distribution/pattern serve as the natural factors and landscape for each city/region, respectively. Given the importance of landscapes, we display them separately. We implement three route types: flight, land, and sea. Flight routes connect any two cities, but abrupt changes between dissimilar cities and the significance of capital cities necessitate land and sea routes. Land routes link capital cities within a country, and sea routes connect port cities of different countries. Directly connecting all capitals would clutter the map, so we calculate routes based on minimal cumulative dissimilarity, refining them with a minimum spanning tree algorithm for clarity. These routes allow users to observe pattern transitions between subspaces and track pattern evolution among representative subspaces, clarifying the impact of different dimensions.

V. Subspace-Map System

The prototype system consists of three views (Fig. 4): Map View, Subspace List View, and Map Detail View.

A. Map View

The Map View (Fig. 4b) visualizes the map in Section III-C. Each city is a hexagon, with color and distance indicating

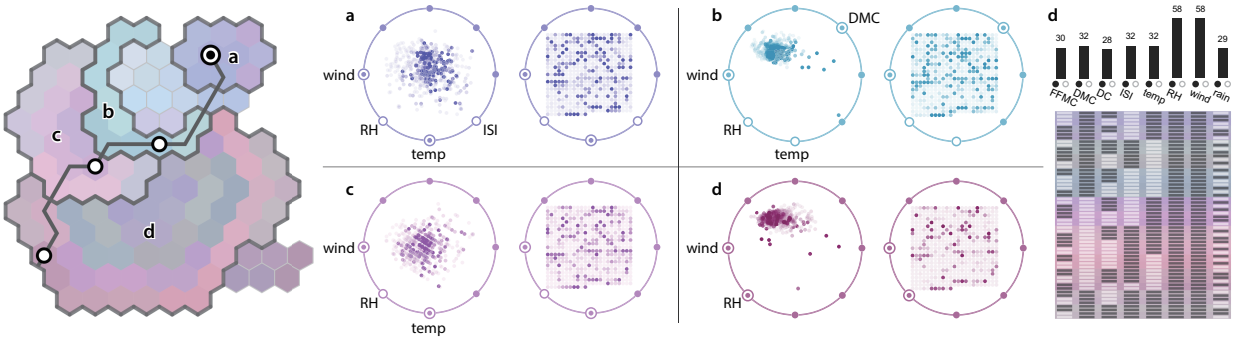


Fig. 5. Forest Fires Data: the analysis of cluster A. A can be divided into four sub-clusters. *RH* and *temp* dominate the clustering at this level. *DMC* and *ISI* also play a part in some sub-clusters. Judging from Subspace List View, *temp* does not stand out in sub-cluster d.

subspace similarity. Clusters and sub-clusters are depicted as countries and provinces, with capitals representing their representatives. Outliers are shown as islands and municipalities. Users can navigate cluster levels by zooming and panning.

At the urban level, natural factors are shown as fan-shaped glyphs (Fig. 4d and Fig. 6 right (a)). Filled fans signify the presence of a dimension, and unfilled fans indicate absence. Transparency highlights shared local-scale dimension patterns among neighboring subspaces (Fig. 6 right (b)). A fan is more opaque if it appears frequently in the subspace's j-neighborhood (see 'Similarity measurement' in Section IV-A).

Users can select different route types for inter-city travel. Flight routes enable direct travel between any cities, while land and sea routes connect capital cities within and across countries, respectively.

B. Subspace List View

The Subspace List View (Fig. 4a) shows dimensional details of selected subspaces. It uses a histogram to depict dimension distribution, with buttons below each bar for filtering subspaces based on dimension inclusion or exclusion. The list beneath the histogram displays subspaces in traversal order. Each list row corresponds to a subspace, with its background color matching the Map View. Black and white boxes denote the presence or absence of dimensions. This color-order integration visually demonstrates each cluster's dimension patterns. Users can select subspaces by clicking on them. The list updates to display cities along the chosen route in travel mode.

C. Map Detail View

The Map Detail View (Fig. 4c) showcases dimension and data patterns of selected clusters. MDS projections depict data distributions of representative subspaces, with point opacity indicating data stability. A more opaque point means its data item is stable across cluster members, suggesting a minimal impact of dimension changes on its neighbors (see 'Data pattern' in Section IV). MDS projections are susceptible to rotation, scaling, and translation. To maintain a consistent mental map, we apply Procrustes transformation [64], aligning different sets of positions to minimize geometric differences. Dimension patterns are indicated on the boundary circle with filled or unfilled circles for commonly included or excluded dimensions, along with their names (Fig. 5).

We also offer a matrix-style alignment for data points, referred to as the stability matrix (Fig. 5). Different matrices display data in the same order for easy visual comparison of patterns. Users can toggle between the matrix and the projection. The projection is suitable for analyzing data item similarities, whereas the matrix, avoiding overlap, is more effective for assessing data item stability—whether they undergo significant changes or stay consistent.

At national or provincial levels, this view presents data and dimension patterns of clusters or sub-clusters. At the urban level, it displays details of the selected subspace. In travel mode, it shows the starting, current, and ending subspaces.

D. Exploration Workflow

The three views are closely integrated for effective subspace exploration (Fig. 2). The Map View serves as the primary exploration entry, initially showing an overview of subspace clusters. Users can delve into provincial or urban levels by double-clicking a cluster or using the plus button in the scale panel (Fig. 4e). At the urban level, glyphs depicting local-scale dimension patterns become visible. The Subspace List View and Map Detail View adjust to these level changes. The Subspace List View shows the dimensions of each selected subspace and their frequency, with options to select subspaces or filter them based on dimension inclusion or exclusion. The Map Detail View reveals data patterns of clusters and dominant dimension patterns. Users can switch between projection and stability matrix to focus on similarity relationships or data item stability using the switch button.

Additionally, users can explore pattern transitions across subspaces in air-travel or ground-travel mode, accessible via the exploration mode panel (Fig. 4e). The air-travel mode enables travel between any subspaces, while the ground-travel mode connects capital cities through land and sea routes. Travel is initiated by selecting start and end subspaces.

VI. Evaluation

This section demonstrates Subspace-Map's effectiveness through two case studies, a user study, and a qualitative comparison with state-of-the-art approaches.

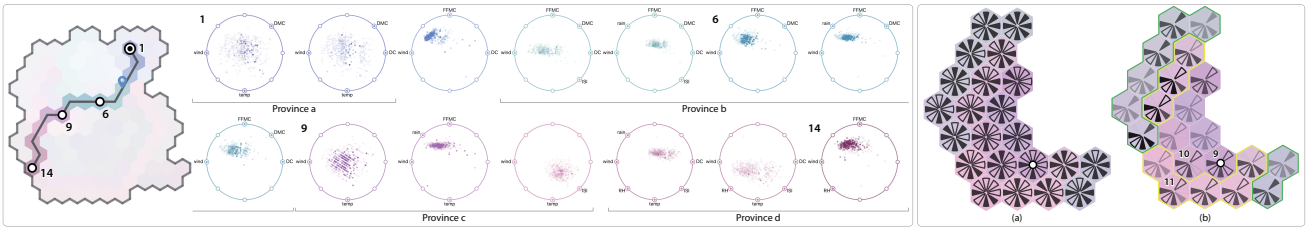


Fig. 6. Analysis of Forest Fires Data. Left: pattern transitions in cluster A. The route goes through 14 cities across 4 provinces, allowing to browse the gradual change of patterns. It is worth noticing that subspace 9 to 11 present quite different data patterns. Right: local patterns within sub-cluster A-c. (a) For each subspace, we show its dimensions in a glyph style. However, it is hard to find trends visually. (b) Therefore, we highlight shared dimension patterns in different neighborhoods for better perception. Subspace 9 to 11 shows different patterns, which explains their data diversity.

A. Case 1: Forest Fires Data

The Forest Fires dataset [65] contains 517 records from Montesinho Natural Park, Portugal, and was used by Jäckle et al. [5] for a case study. The park is divided into 72 regions, with each record detailing the date and region of a forest fire along with eight environmental factors: *Temperature (temp)*, *Relative Humidity (RH)*, *Wind speed (Wind)*, *Rainfall (Rain)*, *Fine Fuel Moisture Code (FFMC)*, *Duff Moisture Code (DMC)*, *Drought Code (DC)*, and *Initial Spread Index (ISI)*. These factors are part of the Forest fire Weather Index (FWI), used by experts to estimate wildfire risk. Excluding all 1D subspaces, we identified a total of 247 subspaces. This case is also mentioned in the video.

Fig. 4 shows an overview of the map, showing two major clusters (countries). From the Map Detail View (Fig. 4c), *Wind* emerges as the dominant dimension at the national level. Most subspaces in country A include *Wind*, while the opposite is true for country B. Comparing their projections, it is evident that country B's subspaces are more prone to forming high-density clusters. Moreover, country A's projection has fewer high-opacity points, indicating less stable data structures.

Curious about the inner division of cluster A, we explore its provincial level, revealing four sub-clusters (Fig. 5). Sub-cluster d is notably larger than the others. The Map Detail View indicates that *RH* and *temp* are key in provincial division. *RH* is prevalent in sub-cluster d but absent in sub-clusters a to c. *temp*, along with *ISI* and *DMC*, differentiates the three smaller sub-clusters. The data structures in sub-clusters b and d show more clustering, likely due to some extreme instances. The Subspace List View shows *RH* and *Wind* as dominant in sub-cluster d, with other dimensions contributing less. This confirms the insights from the Map Detail View.

Before moving on, we delve into the semantics of our findings. In the FWI system, *temp*, *RH*, *Wind*, and *Rain* are natural factors that generate other indicators. *FFMC*, *DMC*, and *DC* reflect water contents at different surface levels, while *ISI*, derived from *FFMC*, shows potential wildfire spread rates. *RH* and *temp* directly impact the three water content indicators, explaining their dominance in provincial-level clustering. *Wind* stands out due to its independence from other dimensions. Portugal's climate, with hot, dry summers and cold, wet winters, means *RH* and *temp* are seasonally correlated but similar across regions. *Wind*, however, is more influenced by topography and less by seasons, making it the most informative dimension as it cannot be deduced from others.

With an understanding of each sub-cluster's features, we explore pattern changes across them via travel routes. We follow a land route through all provincial capitals (Fig. 6 left), visiting 14 subspaces. Subspaces 1 and 2 exhibit high similarity with scattered projections. Subspace 3, an outlier, shows abrupt pattern changes. Subspaces 4 to 8, in sub-cluster b, demonstrate gradual data pattern shifts. Subspaces 9 to 11 in sub-cluster c are less consistent. Finally, subspaces 12 to 14 illustrate a transition from scattered to clustered data patterns.

To understand why subspaces 9 to 11 exhibit varied data patterns, we zoom into province c's urban level (Fig. 6 right). Fig. 6 right (a) displays dimensions of all subspaces in glyph form, but discerning trends is challenging. By highlighting common dimension patterns in each subspace's neighborhood, we achieve clearer visualization in Fig. 6 right (b). This method does not alter the original design but varies the opacity of different dimensions. We observe several dimension patterns within sub-cluster c. Subspaces 9, 10, and 11 exhibit 3 local patterns with varying states of *ISI*, accounting for their data pattern diversity. Two prevalent patterns, highlighted in the figure, originate from divided regions, indicating that some areas are isolated on the map. This limitation in our algorithm will be discussed in Section VII.

B. Case 2: Handwritten Digits Data

The handwritten digits data [66] contains 10,992 digits from 0 to 9, initially gathered for classification tasks [67]. We use its testing set of 3,498 instances. Each digit's trajectory is resampled into 8 equidistant points, creating a 16-dimensional feature vector represented by (x, y) coordinates. In our illustrations, color indicates sampling order (light to dark), and dimensions are labeled F0 through F15. Using SURFING, we identified 315 valuable subspaces out of 65,519 candidates.

The map reveals four distinct countries (Fig. 7a), with a common pattern where even-numbered dimensions (F0, F2, etc.), representing x coordinates, are mostly excluded. SURFING's focus on clustered data suggests that x coordinates do not effectively group trajectories of the same digit, likely due to significant variation in their x values. This indicates that digit recognition is more influenced by y-axis patterns, aligning with everyday observations.

The Map Detail View provides limited insight due to cluttered projections without clear digit separation. Therefore, we examine each digit individually to uncover specific details and relationships (Fig. 7b, c, d). Most digits form distinct

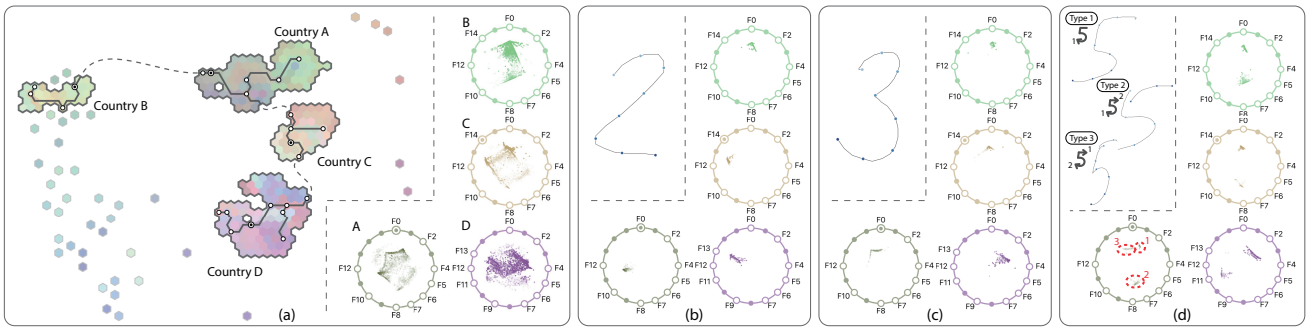


Fig. 7. Handwritten Digits Data: a comprehensive analysis. (a) The subspaces form four clusters featuring mainly even-number dimensions. The handwritten trajectories of most digits (e.g., (b) digit 2 and (c) digit 3) only form one cluster in each country. (d) However, digit 5 has three clusters (highlighted in red dotted circles) in country A because people write digit 5 in three different ways.

clusters under the four subspace settings. For instance, digits 2 and 3 each form a single cluster in every country, whereas digit 5 has two distinct clusters. Notably, digit 5 forms three separate clusters in country A.

To understand digit 5's unusual pattern, we examine its raw trajectories and discover two common writing styles: starting with the horizontal stroke (type 1) or ending with it (type 2) (Fig. 7d). These styles result in distinctly different y-axis patterns, explaining the two-cluster phenomenon. However, this does not account for the third cluster in country A. Further analysis revealed a variation in the direction of the horizontal stroke (types 1 and 3). This is only captured in country A because it includes F0 (the x coordinate of the starting point), a critical indicator of the stroke's initial direction.

We observe that digit 3's cluster overlaps with one of digit 5's clusters in every country. This is because when 5 is written top-down (types 1 and 3), its y-axis pattern resembles that of 3. However, when including F0, right-to-left stroke versions of 5 (type 1) can be distinguished from 3 (Fig. 7d, red dotted circles). In addition, clusters of 2 and 3 are distinguishable in all countries except B. Analyzing their raw trajectories reveals that the key difference is in their lower halves' direction: 2 ends left-to-right, while 3 is the opposite. B does not capture this due to excluding F14 (the x coordinate of the ending point), crucial for identifying the ending stroke's direction.

C. User Study

We further evaluate Subspace-Map through a user study in analytical scenarios. It has two main objectives: firstly, to determine if Subspace-Map meets the goals outlined earlier (**G1**-**G3** in Section I), and secondly, to compare it with leading methods in subspace visual analysis.

1) Data, Participants, and Settings: Our user study employed the Forest Fires dataset [65] for its prominent subspace features and clear data semantics (Section VI-A). With country A already examined, we designed the experiment to focus on analyzing country B.

We recruited 15 participants (11 males, 4 females, ages 21-26), all undergraduate or graduate students in computer science. Their experience with visual analysis and knowledge of high-dimensional data varied. Significantly, many were unfamiliar with subspace concepts and analysis. This allowed us to assess how effectively intuitive map metaphors can replace complex subspace concepts

The user study was conducted in a separate room with a 27-inch 4K monitor placed on an 80cm high table. Participants sat about 50cm from the monitor, interacting via keyboard and mouse to complete tasks and answer questions.

The experiment had three phases: pre-training, method validation, and method comparison. In pre-training (45 minutes), we briefed participants on the data, subspace concepts, map metaphors, and interfaces of Subspace-Map and its competitor, using a simple dataset for practice. The method validation phase involved participants performing tasks with Subspace-Map, then rating and discussing their experience. In the method comparison phase, they used both systems for similar tasks and compared their usability. The intra-group design allowed participants to experience and evaluate both systems.

2) Method Validation: In the method validation phase, we assessed if Subspace-Map achieves the three goals set in Section I. We designed five tasks based on the Forest Fires dataset (Fig. 4) to align with these goals. Tasks **T1** and **T2** involved identifying countries and provinces and describing their characteristics and relationships (**G1**). Task **T3** asked participants to adjust a city's natural factors to match a given landscape (**G2**). Tasks **T4** and **T5** required comparing cities by following routes at the provincial level and analyzing glyphs at the urban level (**G3**).

For each task, participants rated a related statement on a 5-point Likert scale (1 for strongly disagree, 5 for strongly agree) to indicate their agreement. Open-ended questions followed some tasks to collect detailed feedback. For more information, please refer to the supplementary material.

3) Method Comparison: In the method comparison phase, we evaluated whether Subspace-Map surpasses similar methods in specific analytical tasks. This required selecting comparable subspace visual analysis methods as competitors.

To be 'comparable,' a method should meet, fully or partially, Subspace-Map's main goals (**G1**-**G3** in Section I). We reviewed subspace mining and visualization methods in Section II-B. Some, like those handling only non-axis-aligned subspaces [35] or those with mutually exclusive dimensions [7], may not effectively guide subtle dimensional decisions (**G2**). SMARTexplore [68] lets users define and compare subspaces,

but lacks a comprehensive overview of multiple subspaces (**G1**) and a structured exploration approach (**G3**).

Works closely related to ours include Tatu et al. [4]'s Subspace Search and Visualization and Jäckle et al. [5]'s Pattern Trails. Like Subspace-Map, both methods use subspace search algorithms to focus on interesting subspaces and provide 2D or 1D overviews with aggregation mechanisms for similar subspaces (**G1**). They facilitate understanding the dimension-data interplay (**G2**) and guide exploration with ordered sequences of subspaces (**G3**). However, the interestingness metric of Subspace Search and Visualization might not ensure smooth transitions between subspaces, an essential aspect of our exploration scheduling approach. Besides, we had access to only one prototype of Pattern Trails, developed by a novice engineer. Considering the alignment with our objectives and development constraints, we chose Pattern Trails for our comparative study.

Although Pattern Trails is a suitable comparison, it is important to note differences in task focus and data preprocessing approaches between the two systems. Pattern Trails emphasizes pattern transitions between single or multiple points/clusters of subspaces, categorizing them into seven scenarios to identify the role of dimensions in each. Therefore, while it meets the three goals, its distinct analytical focus might result in varying performance compared to Subspace-Map in certain instances.

The key differences in data preprocessing between Pattern Trails and our system are twofold. Firstly, Pattern Trails employs a unique subspace similarity measure. It projects subspaces into 2D, computes a distance matrix for data items in each subspace, and then linearizes this matrix into a vector representation. Similarity is assessed based on the differences between these vectors. Secondly, Pattern Trails characterizes clusters distinctively, defining a cluster's dimensions as the union of all dimensions in its subspaces and using the corresponding projection to determine the cluster's data pattern.

To assess the performance of Subspace-Map relative to Pattern Trails, we established three hypotheses aligned with our goals:

- **H1:** Subspace-Map provides a more accurate overview of subspaces' characteristics and relationships.
- **H2:** Subspace-Map better reveals the impact of dimensional changes on data patterns and assists users in making dimensional decisions.
- **H3:** Subspace-Map offers a more coherent and understandable path for exploration, with smoother transitions.

H1 is based on Subspace-Map's efficient spatial use in its 2D tiled map overview, which contrasts with Pattern Trails' 1D overlapping display, and its richer information portrayal, such as sub-clusters with distinct colors. **H2** is supported by Subspace-Map's detailed dimensional information, like commonly included/excluded dimensions, which is more precise than Pattern Trails. **H3** is plausible as both methods offer sequences of subspaces ordered by data similarity.

We designed tasks **T6** to **T8** to validate these hypotheses. Considering the unique data preprocessing of both methods and the absence of ground truth in the Forest Fires data, we avoided accuracy-based comparisons. Instead, after using

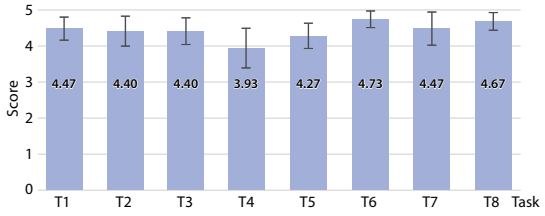


Fig. 8. Participants' ratings of the eight tasks using 5-point Likert scores. Task **T4** was the only one rated slightly below 4, with all others scoring above 4.

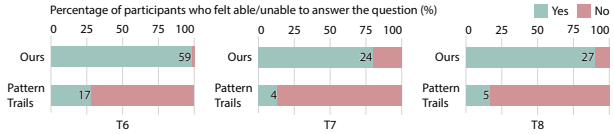


Fig. 9. Participant response statistics on the effectiveness of Subspace-Map and Pattern Trails [5] for tasks **T6** to **T8**. The x-axis shows the percentage of 'yes' and 'no' answers. The number inside each 'yes' bar represents the count of affirmative responses for that task. Task **T6** includes four questions, while tasks **T7** and **T8** each include two questions.

each system, participants answered yes-no questions related to specific data phenomena or the feasibility of analytical tasks. A 'yes' indicated they felt confident answering using the system, and 'no' otherwise. Task **T6**, related to **H1**, involved four questions about subspace cluster characteristics and differences. Task **T7**, addressing **H2**, had two questions on the outcomes of dimensional combinations or adjustments. Task **T8**, for **H3**, included two questions about comparing distant subspaces and planning a path to observe data changes.

Like in the method validation phase, participants rated each task-related statement on a 5-point Likert scale and provided detailed feedback on open-ended questions. For more details, please see the supplementary material.

4) Results: We tracked the time participants took to complete the study, averaging around 40 minutes (Mean = 2429.93 seconds, SD = 655.64 seconds). Analyzing the study's first part, which assessed Subspace-Map's effectiveness, we found that aside from **T4** (scored 3.93), it scored over 4 in tasks **T1** to **T5** (Fig. 8). This indicates Subspace-Map's success in meeting our initial goals.

Under **G1**, participants easily identified different countries (**T1**) and provinces (**T2**) in the Map View, understanding their relationships and characteristics. Feedback highlighted the Map View and Map Detail View as particularly helpful, with color and placement as key visual cues. Map metaphors were praised for their intuitiveness, hierarchical structure, and low communication costs in understanding subspace concepts. However, a few participants noted that these metaphors could add cognitive load due to learning concept mappings.

For **G2**, participants generally grasped the connection between dimensions and data patterns, using these insights to inform their dimensional choices (**T3**).

Regarding **G3**, participants found Subspace-Map effective in selecting their next analytical focus (**T5**), with color and glyph in each grid cell aiding in identifying similar subspaces. Most participants preferred land routes over flight routes for discerning data changes (**T4**), though some critiqued the lack

TABLE I
COMPARISON OF OUR WORK WITH FIVE STATE-OF-THE-ART APPROACHES.

	Similarity measure	Subspace grouping strategy	Subspace layout	Pattern detection strategy
Dimension Projection Matrix/Tree [7]	None	None	Tree and matrix	Observation and manual brushing
Subspace Search and Visualization [4]	Dimension overlap and data topology	Dimension distribution and representative subspace	2D projection	Observation and manual brushing
TripAdvisorND [35]	Dimension vectors	None	2D projection	Observation and manual adjustment
SMARTexplore [68]	None	None	Table	Automatic detection based on template matching
Pattern Trails [5]	Projected distance of data	Dimension union of the cluster and subspace formed by the union	1D projection	Automatic detection based on clusters between adjacent subspaces
Ours - Subspace-Map	Data topology	Representative subspace, data stability, and featured dimensions	Map layout	Automatic detection by 1) dimension and data features of subspaces and clusters and 2) pattern transition routes

of clear patterns or semantics in these changes. One participant suggested that improved route planning or sense-making could address this issue. Notably, 5 participants confused the Map Detail View with RadViz [69]. We consider this a design flaw and will discuss it further in Section VII.

In the study's second part, assessing Subspace-Map against Pattern Trails, participants rated Subspace-Map higher in all three tasks (Fig. 8), echoed by their responses to the questions. Fig. 9 shows more participants felt capable of completing tasks with Subspace-Map. McNemar's test confirmed significant differences for all tasks: $\chi^2 = 42$, $p < .0001$ for **T6**; $\chi^2 = 18.182$, $p < .0001$ for **T7**; and $\chi^2 = 22$, $p < .0001$ for **T8**. These results support **H1**, **H2**, and **H3**.

Participants' feedback identified two key reasons for Pattern Trails' perceived inadequacy in task completion. First, Pattern Trails often shows many subspace clusters without discernible differences due to its cluster representation strategy (see paragraph 5 in Section VI-C3). This approach masks the dimension patterns of clusters, obscuring which dimensions are significant in each cluster and potentially leading to identical representations for different clusters.

Secondly, participants tried to counter repetitive cluster representations in Pattern Trails by increasing the number of displayed subspace clusters or showing all subspaces directly. However, they found that too few clusters still led to repetition, while too many or displaying all subspaces caused overlapping, hindering analysis. Additionally, significant adjustments to cluster numbers made it difficult to track which new clusters or subspaces corresponded to the previous ones.

In summary, the disconnection between different granularity levels, inadequate subspace cluster representation, and overlapping due to the 1D layout in Pattern Trails all reduced participants' confidence in the tool. These issues validate our earlier design decisions.

D. Multi-Perspective Method Comparison

We also aim to compare Subspace-Map with top methods from various perspectives, independent of predefined tasks. We selected five related approaches for comparison: Dimension Projection Matrix/Tree [7], Subspace Search and Visualization [4], TripAdvisorND [35], SMARTexplore [68], and Pattern Trails [5]. The comparison covers four aspects: similarity measure, subspace grouping strategy, subspace layout, and pattern detection strategy (Table I).

Similarity measure. To compare subspaces of varying dimensionalities, TripAdvisorND calculates the Euclidean distance between the dimension vectors of their projections. However, this dimension-focused approach can misrepresent similarity, as small dimension changes might result in significant data pattern differences. It also suffers from information loss due to projection. Pattern Trails, using projected data distance, faces similar information loss issues. Subspace Search and Visualization, on the other hand, assesses data topology resemblance by comparing k-NN lists of data items across subspaces. This method is not dependent on dimensionality and avoids explicit information loss. We adopt a similar measure in our approach.

Subspace grouping strategy. Pattern Trails defines a cluster by its union space, a method that can lead to different clusters having identical representatives and overlooks dimension distribution. Subspace Search and Visualization chooses the representative subspace based on the lowest dimensionality and highest interestingness score, also considering dimension distribution. Our approach offers more comprehensive information, encompassing data stability, dimension stability, and featured dimensions of each cluster.

Subspace layout. Dimension Projection Matrix/Tree employs a hierarchical tree layout for organizing subspaces, enabling dual and recursive exploration. However, it obscures subspace relationships and lacks visual scalability. SMARTexplore uses a table layout, with columns for dimensions, rows for groups/clusters, and color-coded cells for aggregation values. While useful for analyzing patterns like correlations and clusters, it struggles with scalability. The other three methods use projection layouts, offering better overviews but often leading to visual occlusion, hindering detailed subspace analysis. Our map layout, in contrast, is more intuitive, evenly distributes screen space among subspaces, and accurately shows their relationships.

Pattern detection strategy. SMARTexplore and Pattern Trails offer automated exploration features, unlike the other methods. SMARTexplore automatically detects linear corre-

lations, clusters, and outliers using templates against each dimension. Pattern Trails identifies transition patterns based on cluster changes in subspace projections, though this approach is computationally intensive and detects limited patterns. Our method reveals the influence of dimensions on data by providing featured dimensions, data patterns (including projections and stability), and dimension stability within clusters. Additionally, we offer routes to trace pattern transitions smoothly.

VII. Discussion

In this section, we discuss the current limitations and potential enhancements of Subspace-Map.

Subspace retrieval. Subspace clustering algorithms can identify valuable subspaces but often produce redundant results, hindering analysis. Conversely, random sampling preserves data diversity but might miss valuable subspaces. A balanced approach could involve initially using SURFING, then enriching its findings by randomly sampling among non-selected subspaces.

Map layout. Fig. 6 right (b) shows recurring local patterns in different regions, indicating that similar subspaces within a sub-cluster may be separated due to layout algorithm constraints. Our two-level clustering approach allows subspaces similar at both granularities to be in the same sub-cluster or cluster, reflecting subspace similarities at national and provincial levels. However, relationships within a sub-cluster are not further detailed. The current traversal order is based on linear similarity, overlooking potential clustering among subspaces. A potential solution is to implement clustering at a more granular level.

Color encoding. We project the distance matrix into 3D to depict subspace similarity, with each dimension representing an RGB parameter. However, no single parameter in this color space corresponds to a perceptual property, such as hue or brightness [70]. This could lead to inconsistencies between intended and perceived similarities. To improve this, we could use a perceptually-friendly color space like CIELAB [71], or project the matrix into 2D and select colors from a predesigned 2D colormap [72].

Scalability. The primary computational costs lie in dimensionality reduction and k-NN algorithms. Consider a dataset with n data items and d dimensions. MDS projection for each subspace has a time complexity of $\mathcal{O}\left(n^3\right)$, plus $\mathcal{O}\left(d \cdot n\right)$ for k-NN list computation. These costs escalate with increasing d. We address this by using SURFING to reduce subspace numbers and saving results to avoid repetition. Additionally, parallel processing can further enhance speed, as subspace computations are independent.

Scalability in visual representation is influenced by the number of subspaces, dimensions, and data items. We mitigate this by hierarchically organizing subspaces and adjusting hexagon sizes. Yet, our current visualizations, like fan-shaped glyphs and boundary circle icons, become less effective when dimensions exceed a specific number. Alternative designs could include small squares in place of glyphs and moving icons to one side of the boundary circle. These solutions, however, only partly solve scalability issues due to screen space limitations. For instance, small squares must fit within a subspace's hexagon. In the Map Detail View, large data item counts cause scatterplot overlap, which can be remedied with aggregation methods like heatmaps.

Design of the projection. We display the projection in a circular form with boundary icons indicating dimension patterns in our Map Detail View (Fig. 4c). Some participants in our user study, familiar with RadViz, found this design confusing, mistaking it for RadViz [69]. Interestingly, those new to visualization did not experience this confusion. We overlooked the potential for this misconception among users familiar with RadViz, as they might associate the outer dimensions with the inner distribution. We plan to rectify this in future designs to avoid such confusion.

In future work, we aim to incorporate additional analytical techniques and user interfaces. This could involve integrating various subspace clustering algorithms, enabling users to choose the most appropriate one. We also plan to allow users to adjust dimension weights, giving them control over the prominence of specific dimensions in processes like clustering. Additionally, we intend to improve the guidance mechanism, automatically offering users insightful exploration directions.

VIII. Conclusion

We introduce Subspace-Map, a novel method for visualizing and exploring subspaces using map metaphors. It depicts clusters as regions, their representatives as capital cities, and includes routes for tracing pattern transitions between subspaces, etc. We develop a prototype system and demonstrate its effectiveness through two case studies, a user study, and a comparison with state-of-the-art methods. Future enhancements will focus on expanding its analytical capabilities and providing more guidance for exploring subspaces. Our code is available at https://github.com/pkuvis/Subspace-Map-prototype.

Acknowledgments

We thank the anonymous reviewers for their insightful comments. This work was supported by NSFC No. 62272012.

References

[1] H. Kriegel, P. Kröger, and A. Zimek, "Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering," *ACM Transactions on Knowledge Discovery from Data*, vol. 3, no. 1, pp. 1:1–1:58, 2009.

[2] E. Müller, S. Günnemann, I. Assent, and T. Seidl, "Evaluating clustering in subspace projections of high dimensional data," *Proceedings of the VLDB Endowment*, vol. 2, no. 1, pp. 1270–1281, 2009.

[3] H. Kriegel, P. Kröger, and A. Zimek, "Subspace clustering," *WIREs Data Mining and Knowledge Discovery*, vol. 2, no. 4, pp. 351–364, 2012.

[4] A. Tatu, F. Maass, I. Färber, E. Bertini, T. Schreck, T. Seidl, and D. A. Keim, "Subspace search and visualization to make sense of alternative clusterings in high-dimensional data," in *Proceedings of IEEE Conference on Visual Analytics Science and Technology*, 2012, pp. 63–72.

[5] D. Jäckle, M. Hund, M. Behrisch, D. A. Keim, and T. Schreck, "Pattern trails: Visual analysis of pattern transitions in subspaces," in *Proceedings of IEEE Conference on Visual Analytics Science and Technology*, 2017, pp. 1–12.

[6] N. Elmqvist, P. Dragicevic, and J. Fekete, "Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1141–1148, 2008.

[7] X. Yuan, D. Ren, Z. Wang, and C. Guo, "Dimension projection matrix/tree: Interactive subspace visual exploration and analysis of high dimensional data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2625–2633, 2013.

[8] D. T. Nhon and L. Wilkinson, "Scagexplorer: Exploring scatterplots by their scagnostics," in *Proceedings of IEEE Pacific Visualization Symposium*, 2014, pp. 73–80.

[9] A. Inselberg, "The plane with parallel coordinates," *The Visual Computer*, vol. 1, no. 2, pp. 69–91, 1985.

[10] J. Yang, W. Peng, M. O. Ward, and E. A. Rundensteiner, "Interactive hierarchical dimension ordering, spacing and filtering for exploration of high dimensional datasets," in *Proceedings of IEEE Symposium on Information Visualization*, 2003, pp. 105–112.

[11] C. Turkay, A. Lundervold, A. J. Lundervold, and H. Hauser, "Representative factor generation for the interactive visual analysis of high-dimensional data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2621–2630, 2012.

[12] Z. Zhang, K. T. McDonnell, E. Zadok, and K. Mueller, "Visual correlation analysis of numerical and categorical data on the correlation map," *IEEE Transactions on Visualization and Computer Graphics*, vol. 21, no. 2, pp. 289–303, 2015.

[13] J. Seo and B. Shneiderman, "A rank-by-feature framework for interactive exploration of multidimensional data," *Information Visualization*, vol. 4, no. 2, pp. 96–113, 2005.

[14] D. B. Carr, R. J. Littlefield, W. Nicholson, and J. Littlefield, "Scatterplot matrix techniques for large n," *Journal of the American Statistical Association*, vol. 82, no. 398, pp. 424–436, 1987.

[15] M. Sips, B. Neubert, J. P. Lewis, and P. Hanrahan, "Selecting good views of high-dimensional data using class consistency," *Computer Graphics Forum*, vol. 28, no. 3, pp. 831–838, 2009.

[16] L. Wilkinson, A. Anand, and R. L. Grossman, "Graph-theoretic scagnostics," in *Proceedings of IEEE Symposium on Information Visualization*, 2005, pp. 157–164.

[17] M. Sedlmair, A. Tatu, T. Munzner, and M. Tory, "A taxonomy of visual cluster separation factors," *Computer Graphics Forum*, vol. 31, no. 3pt4, pp. 1335–1344, 2012.

[18] A. Dasgupta and R. Kosara, "Pargnostics: Screen-space metrics for parallel coordinates," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 6, pp. 1017–1026, 2010.

[19] S. Johansson and J. Johansson, "Interactive dimensionality reduction through user-defined combinations of quality metrics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 993–1000, 2009.

[20] A. Tyagi, T. Estro, G. Kuenning, E. Zadok, and K. Mueller, "Pc-expo: A metrics-based interactive axes reordering method for parallel coordinate displays," *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, no. 1, pp. 712–722, 2023.

[21] E. Bertini, "Quality metrics in high-dimensional data visualization: An overview and systematization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2203–2212, 2011.

[22] M. Behrisch, M. Blumenschein, N. W. Kim, L. Shao, M. El-Assady, J. Fuchs, D. Seebacher, A. Diehl, U. Brandes, H. Pfister, T. Schreck, D. Weiskopf, and D. A. Keim, "Quality metrics for information visualization," *Computer Graphics Forum*, vol. 37, no. 3, pp. 625–662, 2018.

[23] A. Abid, V. K. Bagaria, M. J. Zhang, and J. Y. Zou, "Contrastive principal component analysis," *CoRR*, vol. abs/1709.06716, 2017.

[24] T. Fujiwara, O. Kwon, and K. Ma, "Supporting analysis of dimensionality reduction results with contrastive learning," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 45–55, 2020.

[25] J. Knittel, A. Lalama, S. Koch, and T. Ertl, "Visual neural decomposition to explain multivariate data sets," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 1374–1384, 2021.

[26] K. Wongsuphasawat, D. Moritz, A. Anand, J. D. Mackinlay, B. Howe, and J. Heer, "Voyager: Exploratory analysis via faceted browsing of visualization recommendations," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 1, pp. 649–658, 2016.

[27] K. Wongsuphasawat, Z. Qu, D. Moritz, R. Chang, F. Ouk, A. Anand, J. D. Mackinlay, B. Howe, and J. Heer, "Voyager 2: Augmenting visual analysis with partial view specifications," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 2017, pp. 2648–2659.

[28] A. Sarvghad, M. Tory, and N. Mahyar, "Visualizing dimension coverage to support exploratory analysis," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 21–30, 2017.

[29] C. Turkay, P. Filzmoser, and H. Hauser, "Brushing dimensions - A dual visual analysis model for high-dimensional data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 2591–2599, 2011.

[30] S. Cheng and K. Mueller, "The data context map: Fusing data and attributes into a unified display," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 1, pp. 121–130, 2016.

[31] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," in *Proceedings of ACM SIGMOD International Conference on Management of Data*, 1998, pp. 94–105.

[32] K. Kailing, H. Kriegel, P. Kröger, and S. Wanka, "Ranking interesting subspaces for clustering high dimensional data," in *Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases*, vol. 2838, 2003, pp. 241–252.

[33] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 226–231.

[34] M. Hund, I. Färber, M. Behrisch, A. Tatu, T. Schreck, D. A. Keim, and T. Seidl, "Visual quality assessment of subspace clusterings," in *KDD Workshop on Interactive Data Exploration and Analytics (IDEA'16)*, 2016, pp. 53–62.

[35] J. E. Nam and K. Mueller, "Tripadvisor$^{\text{n-d}}$: A tourism-inspired high-dimensional space exploration framework with overview and detail," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 2, pp. 291–305, 2013.

[36] A. Skupin and B. Buttenfield, "Spatial metaphors for visualizing information spaces," 1997, available in CiteSeerX database.

[37] S. I. Fabrikant and B. P. Buttenfield, "Formalizing semantic spaces for information access," *Annals of the Association of American Geographers*, vol. 91, no. 2, pp. 263–280, 2001.

[38] A. Skupin and S. I. Fabrikant, "Spatialization methods: A cartographic research agenda for non-geographic information visualization," *Cartography and Geographic Information Science*, vol. 30, no. 2, pp. 99–119, 2003.

[39] A. Skupin and S. I. Fabrikant, "Spatialization," *The handbook of geographic information science*, pp. 61–79, 2007.

[40] E. R. Gansner, Y. Hu, and S. G. Kobourov, "Gmap: Visualizing graphs and clusters as maps," in *Proceedings of IEEE Pacific Visualization Symposium*, 2010, pp. 201–208.

[41] D. Mashima, S. G. Kobourov, and Y. Hu, "Visualizing dynamic data with maps," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 9, pp. 1424–1437, 2012.

[42] C. Ma, Y. Liu, G. Zhao, and H. Wang, "Visualizing and analyzing video content with interactive scalable maps," *IEEE Transactions on Multimedia*, vol. 18, no. 11, pp. 2171–2183, 2016.

[43] B. Jacobsen, M. Wallinger, S. G. Kobourov, and M. Nöllenburg, "Metrosets: Visualizing sets as metro maps," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 1257–1267, 2021.

[44] P. Rottmann, M. Wallinger, A. Bonerath, S. Gedicke, M. Nöllenburg, and J. Haunert, "Mosaicsets: Embedding set systems into grid graphs," *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, no. 1, pp. 875–885, 2023.

[45] S. Chen, S. Chen, Z. Wang, J. Liang, X. Yuan, N. Cao, and Y. Wu, "D-map: Visual analysis of ego-centric information diffusion patterns in social media," in *Proceedings of IEEE Conference on Visual Analytics Science and Technology*, 2016, pp. 41–50.

[46] S. Chen, S. Li, S. Chen, and X. Yuan, "R-map: A map metaphor for visualizing information reposting process in social media," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 1204–1214, 2020.

[47] M. Hogräfer, M. Heitzler, and H. Schulz, "The state of the art in map-like visualization," *Computer Graphics Forum*, vol. 39, no. 3, pp. 647–674, 2020.

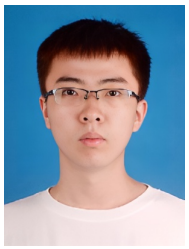
[48] R. P. Biuk-Aghai, C. Pang, and F. H. H. Cheang, "Visualization of large category hierarchies," in *Proceedings of the 2011 Visual Information Communication - International Symposium*, 2011, p. 2.

[49] A. Dieberger and A. U. Frank, "A city metaphor to support navigation in complex information spaces," *Journal of Visual Languages & Computing*, vol. 9, no. 6, pp. 597–622, 1998.

[50] H. Couclelis, "Worlds of information: The geographic metaphor in the visualization of complex information," *Cartography and Geographic Information Systems*, vol. 25, no. 4, pp. 209–220, 1998.

[51] L. Nachmanson, R. Prutkin, B. Lee, N. H. Riche, A. E. Holroyd, and X. Chen, "Graphmaps: Browsing large graphs as interactive maps," in

Proceedings of Graph Drawing and Network Visualization, 2015, pp. 3–15.
[52] R. Preiner, J. Schmidt, K. Krösl, T. Schreck, and G. Mistelbauer, "Augmenting node-link diagrams with topographic attribute maps," *Computer Graphics Forum*, vol. 39, no. 3, pp. 369–381, 2020.
[53] B. Grünbaum and G. C. Shephard, "Tilings by regular polygons," *Mathematics Magazine*, vol. 50, no. 5, pp. 227–247, 1977.
[54] B. Grünbaum and G. C. Shephard, *Tilings and patterns*. Courier Dover Publications, 1987.
[55] R. G. Cano, K. Buchin, T. Castermans, A. Pieterse, W. Sonke, and B. Speckmann, "Mosaic drawings and cartograms," *Computer Graphics Forum*, vol. 34, no. 3, pp. 361–370, 2015.
[56] E. S. Spelke, "Principles of object perception," *Cognitive science*, vol. 14, no. 1, pp. 29–56, 1990.
[57] S. Jahirabadkar and P. Kulkarni, "Scaf - an effective approach to classify subspace clustering algorithms," *International Journal of Data Mining & Knowledge Management Process*, vol. 3, no. 2, pp. 69–86, 2013.
[58] K. Kailing, H. Kriegel, and P. Kröger, "Density-connected subspace clustering for high-dimensional data," in *Proceedings of the Fourth SIAM International Conference on Data Mining*, 2004, pp. 246–256.
[59] C. Baumgartner, C. Plant, K. Kailing, H. Kriegel, and P. Kröger, "Subspace selection for clustering high-dimensional data," in *Proceedings of the 4th IEEE International Conference on Data Mining*, 2004, pp. 11–18.
[60] T. F. Cox and M. A. Cox, *Multidimensional Scaling*. Chapman and Hall/CRC, 2000.
[61] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of Machine Learning Research*, vol. 9, no. 11, 2008.
[62] M. Espadoto, R. M. Martins, A. Kerren, N. S. T. Hirata, and A. C. Telea, "Toward a quantitative survey of dimension reduction techniques," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 3, pp. 2153–2173, 2021.
[63] S. Ingram, T. Munzner, and M. Olano, "Glimmer: Multilevel MDS on the GPU," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 2, pp. 249–261, 2009.
[64] J. C. Gower and G. B. Dijksterhuis, *Procrustes Problems*. Oxford University Press, 2004.
[65] P. Cortez and A. Morais, "A data mining approach to predict forest fires using meteorological data," in *Proceedings of the 13th Portuguese Conference on Artificial Intelligence*, 2007, pp. 512–523.
[66] D. Dua and C. Graff, "UCI machine learning repository," 2017.
[67] F. Alimoglu and E. Alpaydin, "Combining multiple representations and classifiers for pen-based handwritten digit recognition," in *Proceedings of the Fourth International Conference on Document Analysis and Recognition*, vol. 2, 1997, pp. 637–640.
[68] M. Blumenschein, M. Behrisch, S. Schmid, S. Butscher, D. R. Wahl, K. Villinger, B. Renner, H. Reiterer, and D. A. Keim, "Smartexplore: Simplifying high-dimensional data analysis through a table-based visual analytics approach," in *Proceedings of IEEE Conference on Visual Analytics Science and Technology*, 2018, pp. 36–47.
[69] P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley, "Dna visual and analytic data mining," in *Proceedings of the 8th IEEE Visualization Conference*, 1997, pp. 437–441.
[70] A. Zeileis, K. Hornik, and P. Murrell, "Escaping rgbland: Selecting colors for statistical graphics," *Computational Statistics & Data Analysis*, vol. 53, no. 9, pp. 3259–3270, 2009.
[71] J. Schanda, *Colorimetry: understanding the CIE system*. John Wiley & Sons, 2007.
[72] M. Steiger, J. Bernard, S. Thum, S. Mittelstädt, M. Hutter, D. A. Keim, and J. Kohlhammer, "Explorative analysis of 2 d color maps," in *Proceedings of the 23rd International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, 2015, pp. 151–160.



Jincheng Li received Bachelor's Degree in electronic information engineering from Central South University in 2018. He is now a Ph.D. student at the School of Intelligence Science and Technology, Peking University. His research interests include high-dimensional data visualization and decision-making with visualizations.

Chufan Lai is a research associate at the Technology and Engineering Center for Space Utilization, Chinese Academy of Science. His research interests include high-dimensional data visualization, deep learning-driven visualization, and AI-assisted scientific data analysis. Chufan received his Ph.D. degree in computer science from Peking University.

Xiaoru Yuan received a BS degree in computer science and a BA degree in law from Peking University in 1997 and 1998, respectively. In 2005 and 2006, he received an MS degree in computer engineering and a Ph.D. degree in computer science from the University of Minnesota, Twin Cities. He is now a professor at Peking University in the Laboratory of Machine Perception (MOE). His primary research interests lie in scientific visualization, information visualization, and visual analytics, emphasizing large data visualization, high dimensional data visualization, graph visualization, and novel visualization user interface. He is a senior member of the IEEE.