RSVP for VPSA : A Meta Design Study on Rapid Suggestive Visualization Prototyping for Visual Parameter Space Analysis

Manfred Klaffenboeck, Michael Gleicher, Johannes Sorger, Michael Wimmer, and Torsten Möller
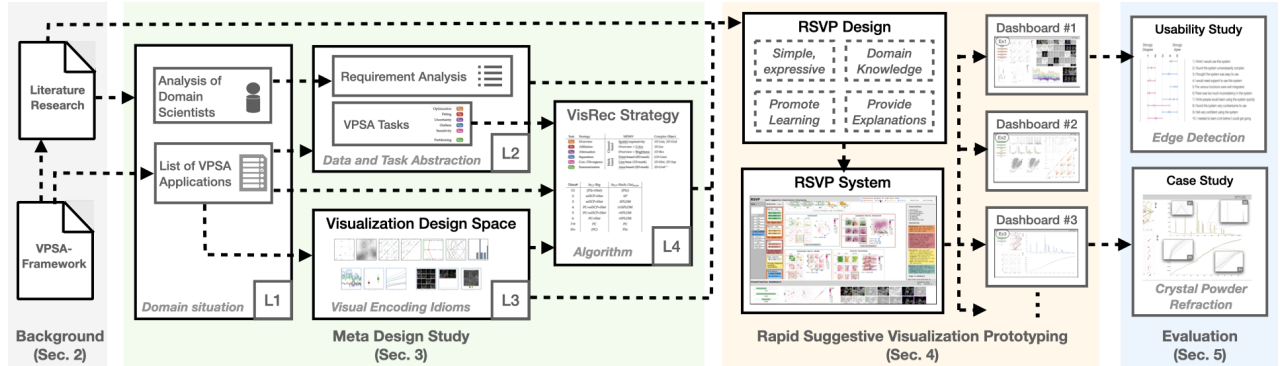

Fig. 1: Overview of our overall process to design a mixed-initiative system that provides domain scientists with the opportunity to perform VPSA without the need for a visualization expert. We conducted a *Meta Design Study* by surveying existing VPSA literature and extracting relevant information. This procedure allowed us to analyze *requirements*, extract a *visualization design space*, and devise a *task-oriented VisRec strategy*. We implemented our findings in RSVP, the *Rapid Suggestive Visualization Prototyping* system. It enables domain scientists to rapidly create and experiment with visualization dashboards tailored to their specific models and data. We externally *evaluated* RSVP's efficiency through a usability study and a real-world case study.

Abstract—Visual Parameter Space Analysis (VPSA) enables domain scientists to explore input-output relationships of computational models. Existing VPSA applications often feature multi-view visualizations designed by visualization experts for a specific scenario, making it hard for domain scientists to adapt them to their problems without professional help. We present RSVP, the Rapid Suggestive Visualization Prototyping system encoding VPSA knowledge to enable domain scientists to prototype custom visualization dashboards tailored to their specific needs. The system implements a task-oriented, multi-view visualization recommendation strategy over a visualization design space optimized for VPSA to guide users in meeting their analytical demands. We derived the VPSA knowledge implemented in the system by conducting an extensive meta design study over the body of work on VPSA. We show how this process can be used to perform a data and task abstraction, extract a common visualization design space, and derive a task-oriented VisRec strategy. User studies indicate that the system is user-friendly and can uncover novel insights.

Index Terms—input-output model, literature analysis, mixed-initiative system, unobtrusive visualization recommendation, user study

---◆---

1 INTRODUCTION

The use of input-output-oriented models is ubiquitous in modern research, spanning problem domains from climate research [1], [2] to complex engineering tasks [3], [4], from parametric design problems [5], [6] to machine learning and deep learning [7], [8]. To comprehensively understand a

- *M. Klaffenboeck and M. Wimmer are with the Institute of Visual Computing and Human Centered Technology at TU Wien, Austria; Email: {lastname}@cg.tuwien.ac.at.*

- *J. Sorger was with the Complexity Science Hub, Vienna, Austria.*

- *M. Gleicher is with the Department of Computer Sciences at the University of Wisconsin - Madison. Email: gleicher@cs.wisc.edu*

- *T. Möller is with the Faculty of Computer Science at the University of Vienna and the Research Network Data Science at Uni Wien. Email: torsten.moeller@univie.ac.at*

model's behavior, it is crucial to go beyond mere trial and error testing of parameter configurations. A more holistic approach performed by some domain experts is to systematically sample the model, relating those sampled input parameters to the respective results and analyzing them together visually. This process is known as *Visual Parameter Space Analysis (VPSA)* [9].

VPSA provides powerful means for scientists to investigate the behavior of complex models under different conditions and reveal patterns and relationships inherent in the model that might otherwise stay hidden. It can aid domain scientists in identifying optimal parameter settings, exploring the underlying model uncertainty in different regions, and finding model boundaries for input parameterizations based on their corresponding outputs.

Despite promising evidence of its usefulness [9], VPSA has not been adopted by the broader research community as

a general problem-solving strategy. A reason for this might be the lack of a visualization tool that allows domain experts to apply this type of visual analysis to their input-output models. VPSA applications designed to solve high-level problems often comprise multiple visualizations capable of performing multi-dimensional analysis. Designing such a multi-view visualization with its various tradeoffs often exceeds the visualization knowledge of domain scientists. In some cases, domain scientists and visualization experts collaborate in design studies to develop customized solutions for specific problems. However, this process requires time, resources, and mutual interest, which may exclude users seeking to utilize VPSA for less significant problems [10].

We introduce a mixed initiative system [11] called RSVP (*Rapid Suggestive Visualization Prototyping*) which encodes VPSA knowledge and guides users in designing visualization dashboards for their data and needs. To gather and implement the necessary knowledge, we developed a meta-design strategy to extract tacit knowledge from VPSA papers and transform it into explicit knowledge [12]. We refer to this approach as a *meta design study*. The study yielded a comprehensive requirement analysis, a VPSA-specific visualization design space, and a visualization recommendation (VisRec) strategy for creating multi-view dashboards to solve typical VPSA tasks. RSVP implements these findings, making an otherwise complex and time-consuming task actionable and comprehensible for non-visualization experts. The system aims to be focused, transparent, and to promote learning to *minimize the cost* of using it and *build trust* in it – two factors we identified as latent needs of domain scientists when adopting a new visualization system.

Outline & Contributions: Our overall approach is illustrated in Figure 1. We introduce important concepts and related work (section 2), followed by our main contributions:

- We devised and conducted a *Meta Design Study* over the body of knowledge on VPSA, complementing and extending previous work in the field [9]. The artifacts from this process include a **comprehensive VPSA requirement analysis**, an **expressive VPSA visualization design space**, and a **practical VisRec strategy** (section 3).
- The *Rapid Suggestive Visualization Prototyping (RSVP)* system, which incorporates the design knowledge from the meta-design study into a user-friendly drag-and-drop interface for creating multi-view dashboards (section 4).
- A *multi-faceted evaluation*, which includes a **qualitative result inspection** (QRI, (subsection 4.3)), a **usability study**, and **two real-world case studies** ((section 5)) to demonstrate the validity and efficacy of our approach.

We discuss findings, design aspects, limitations, and future research opportunities in section 6 and present our conclusion in section 7.

2 Background

First, we introduce VPSA using a practical example. Next, we define the term *meta design study* and discuss *domain experts* as a general user group. Finally, we will review related work on visualization recommendation, exploration, and construction tools.

2.1 VPSA - Introduction

VPSA is a general approach to perform input-output-oriented, multi-dimenional analysis. The overall VPSA workflow is presented in Figure 2. We use a well-known edge detection algorithm [13] as a **running example** to introduce important VPSA concepts and terminology. We opted to use a generic image to avoid potential domain complexities. However, image segmentation, similar to the one presented, is often necessary in biological [14] and medical applications [15], as well as in material sciences [16].

Running Example: The edge detection model (see Figure 2a) takes the unsegmented image and three numerical parameters (*low*, *high*, σ) for internal thresholds as inputs. It outputs a binary contour outline (*co*) and two statistical parameters (*sep* and *wep*). For analytical purposes, the contour outline gets transformed into a 1D projection and compared to the same projection of a hand-drawn *"Ground Truth"* image, resulting in the difference between these two transformed contour outlines (Δ_{tco}). The Goodness-of-Fit between the contour outline and the ground truth (χ^2_{co}) and between their respective 1D projections ($\chi^2_{\Delta_{tco}}$) is measured using Pearson's Chi-square test.

Concepts and Terminology: Figure 2a is organized in the form of the **data flow model** for VPSA [9]. The three input parameters are control parameters (In_{CP}), which the user can directly manipulate. The "Original Image" is an environmental parameter (In_{EP}), often outside the user's direct control, meaning if a specific image needs to be segmented, it cannot be freely exchanged for another. The segmented image (*co*) and the statistical parameters (*sep* and *wep*) are direct outputs (Out_{dir}). These may need further processing for analytical efficiency, resulting in derived outputs (Out_{der}). This example illustrates how VPSA handles various data types, classified broadly into *multi-dimensional/multi-variate (MDMV)* and *complex objects*. In our example, all numerical parameters are MDMV, while the segmented image (*co*) and its transformed 1D-representation (Δ_{tco}) are examples of complex objects A complex object is a semantic unit that can not be described with a single quantitative/ordinal/categorical variable without losing information. Control parameters affect the segmentation process and require adjustment for the desired model output. A non-VPSA approach achieves this through trial-and-error, where the user adjusts parameters after inspecting unsatisfactory results. VPSA replaces this tedious (manual sampling) process by using (systematic) **sampling** to vary control parameters. *Regular* or *stochastic* sampling is often used for this purpose, and the corresponding inputs and outputs are stored in a single data table (Figure 2b). The columns of the data table reprise the *dimensions* to be analyzed, and each row represents the inputs and outputs of a single *run*. Data organized in such a way can then be analyzed visually (like in the example dashboard in Figure 2c). Users adopting VPSA typically aim to perform a certain set of *analysis tasks*, such as finding optimal parameterizations, identifying potential outliers, or partitioning the data. Table 1 provides an overview of typical VPSA tasks, as outlined by Sedlmair

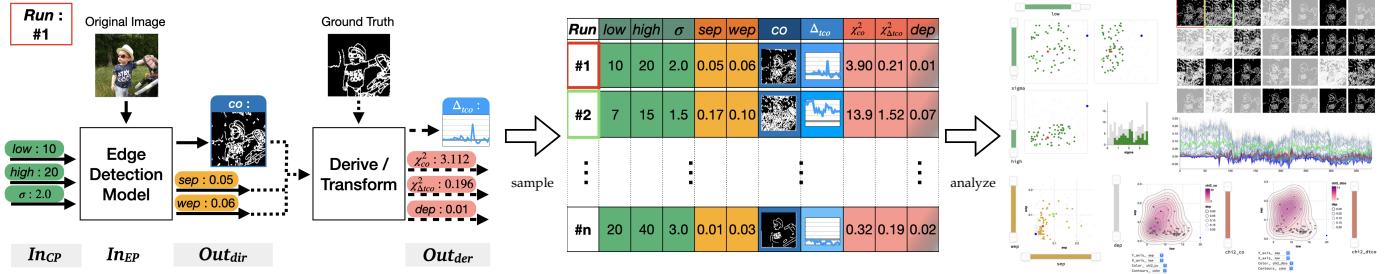| (a) Practical Dataflow Example | (b) Sampled data | (c) Interactive VPSA Session |

Fig. 2: Overview of the VPSA workflow. (a) depicts the dataflow model for VPSA, using an edge detection algorithm as a practical example. The model gets run multiple times using some kind of sampling strategy. The **inputs**, **direct outputs**, **derived outputs**, and resulting **complex objects** (shades of blue) from this process are gathered and stored in a data structure, such as a data table (b), which serves as the data source for the visual analysis process (c). The colors used for the visualizations in (c) are designed to match the colors in (a) and (b) in order to show the main data variable they are trying to encode.

et al. [9]. Please note that the colors used to classify the individual tasks are different from the colors used for inputs and outputs in Figure 2. To perform these tasks, VPSA users need to navigate a multi-dimensional problem space. The most common **navigation strategy** is the *global-to-local approach*, where users start with an overview of the data and then explore details. Less commonly used strategies include the *local-to-global* approach, where users start with a specific sample point and explore alternatives, and the *steering* approach, where users adjust parameters during simulation runtime.

2.2 Meta Design Study

Traditional design studies [10] and data-driven design studies [17] require the early collaboration with domain scientists to gather domain knowledge. In contrast, we apply a design approach that synthesizes this kind of information from written reports of existing design studies. Since this strategy shares many similarities with *meta analysis* [18], we call this a **meta design study**. A meta design study is suitable when the goal is to develop a more general tool that serves the needs of a wider group of users within a certain problem domain. A meta design study offers a principled approach to performing requirements analysis and extracting a visualization design space from analyzed applications and reports. Additionally, it provides a mechanism for deriving a data and task-oriented visualization recommendation strategy over the extracted design space. Practical details will be further explained in section 3.

Meta design studies follow Munzner's nested model for visualization design [19] in a top-down manner [20]: Gathering and analyzing the papers is learning about the *domain situation* L1. The requirement extraction process includes

data and task abstraction L2. Extracting the visualization design space from the literature is acquiring knowledge about *visual encoding idioms* L3, and deriving a visualization recommendation strategy over the visualization design space operates on an *algorithmic level* L4.

Since there is no immediate interaction with **domain scientists**, we extract general knowledge about this user group from existing literature in a similar fashion. Several sources state that domain experts often have little time and prefer visualizations capable of providing immediate data insights [21], [22]. Somewhat contrary to this statement, domain scientists also seem to want to explore their various options [23], [24]. It is beneficial for domain scientists to use familiar data when learning about unknown visualizations [**?**], [25], [26]. Unlike casual users, domain scientists analyze visualizations in a more structured way [27], and they are skilled in transferring their domain knowledge from familiar to unfamiliar visualizations [28]. Furthermore, showing transitions from simple to more complex visualizations fosters an understanding of the latter [29]. However, domain scientists are a heterogeneous user group with different backgrounds, interests, and levels of visualization literacy, which is why they should be treated similarly to visualization novices [30], [31]. From a high-level perspective, these guidelines try to **minimize the cost** [32] of using a visualization tool.

Furthermore, domain experts do not like being told what the (supposedly) best solution is without being able to explore alternative options [4], [24]. We interpret this behavior as a necessity for those experts to build a high enough **level of trust** [33] in a proposed solution. We deem this a crucial factor when designing a general-purpose tool intended to be used by domain scientists that utilizes a recommendation strategy.

2.3 Related Work

In this section, we present related work in terms of visualization recommendation, exploration, and construction.

Visualization Recommendation (VisRec) is a versatile and extensively researched field [34]. Here, we provide background information on various aspects of VisRec that are important for our further discussion.

Data-oriented VisRec: APT [35] was amongst the first tools to provide visualization recommendations. It recom-

TABLE 1: List of typical VPSA-Tasks according to the conceptual framework by Sedlmair at al. [9]

Task		Description
Optimzation	T_{opt}	Find the best parameter setting
Fitting	T_{fit}	Find where actual model data occurs
Uncertainty	T_{unc}	Determine the reliability of the output
Outliers	T_{out}	Find odd or special outputs
Sensitivity	T_{sens}	Identify input regions with high or low impact on the output
Partitioning	T_{part}	Identify different types of model behavior

mended visualizations based on their *expressiveness* (how well does it show the data and only the data) and their (perceptual) *effectiveness*. Although the underlying rules may have been reviewed and refined since then, the basic principles of expressiveness and effectiveness are still vital in modern data-oriented visualization recommendation approaches (e.g. [36], [37], [38]). However, basing recommendations mainly on expressiveness and perceptual effectiveness has its limits and does not necessarily help in solving specific tasks [39]. Our VisRec strategy implicitly covers expressiveness and effectiveness but also takes the task-based effectiveness of visualizations into account.

Task-oriented VisRec: Early work in task-oriented VisRec [40], [41], [42], [43] focused on creating composited graphics for *presentational* purposes and did not necessarily help overcome analytical gaps [44]. Our work focuses on visualizations and strategies for *exploratory* and *confirmatory* data analysis. Modern insight-oriented [45], [46] or task-oriented [47] VisRec systems support a wide range of analytical tasks [44]. Interestingly, their general VisRec approach is strikingly similar. Each task is either associated with a fixed visualization or with visual features that are supposed to support this task. However, the process of finding visualizations / visual features best supporting the respective tasks in the mentioned papers is not explicitly reported. Draco [38] can learn weights to incorporate task-oriented efficiency in the recommendations. Unfortunately, this requires an extensive corpus of labeled training data in the form of ranked examples, even for learning only a minimal number of very general tasks. Furthermore, Draco only supports MDMV visualizations. Our VisRec approach bases its task-oriented recommendations on visual features which we extracted from the meta design study. Additionally, our approach can not only handle MDMV visualizations but also complex object visualizations.

Multi-view VisRec: Solving high-level analysis tasks often requires the combination of multiple visualizations into a coherent whole. VizDeck [48] allows for combining recommended visualizations from different sources into a dashboard, but it does not provide any guidance on how to combine them effectively. DeepEye [49] and MultiVision [50] recommend combinations for multi-view visualizations, but their focus is on general-purpose dashboards and less on solving specific analysis tasks.

Transparency and Explainability in VisRec: The importance of explanations in recommender systems is well established [51] – a fact often ignored in VisRec systems [52]. Some systems (e.g., [49], [53]) report the internal ranking scores to the users, but those most likely have no meaning to them. KG4Vis [54] manages to explain why a visualization was recommended but is currently limited to visualizations encoding a maximum of two attributes. Our literature-based observations for task-based efficiency follow generally accepted visualization design rules and can be easily communicated to the users.

VisRec Knowledge Encoding: Various methods exist for encoding VisRec knowledge, including machine [55] or deep learning [49] for rule acquisition, knowledge graphs [55], constraint-based rule definition with answer set programming [38], and ontologies [56], [57], [58]. Classical approaches [35], [41] employ hard-coded rules, an approach

also taken by CompassQL [36], the recommendation strategy behind Voyager [37], [59]. For simplicity, VisRec rules encoded in RSVP are currently also hard-coded.

Visualization Exploration: *Spreadsheet-like interfaces* for visualization exploration [60] have been used to perform parameter space exploration and present the results to the user [61]. Voyager [37], [59] leverages visualization recommendations to present the user with a single optimal solution for a given data state. One can argue, though, that it is important to present the same data through different perspectives, so the user can understand the data better, avoid possible misconceptions [62], and foster the learning of unknown or less familiar visualizations [63]. Van den Elzen [63] proposed an interface combining small multiples and large singles to show variations of the current state space via various visual mappings. We build upon this idea to combine parameter space exploration with an unobtrusive visualization recommendation strategy for coordinated multiple views.

Multi-view Visualization Construction: Improvise [64], ComVis [65], and Visplore [66] are frameworks that allow creating applications suited for VPSA, but all of them require at least some level of programming, and they do not immediately encode visualization knowledge for domain experts. Keshif [67] is a tool aimed at visualization novices to perform data exploration via multi-view applications. It favors aggregated views to explore big data tables. The analysis of input-output models, however, often requires a focus on visualizations where the individual items are directly encoded and not abstracted away [4]. Tableau [68], [69] provides various visualizations suitable for VPSA but has limited support for complex objects and lacks sufficient guidance for choosing visualizations capable of solving high-level VPSA tasks.

3 VPSA META DESIGN STUDY

This section outlines the procedure and outputs of our *meta design study*.

3.1 Method (L1)

We began by creating a *coding table*, a common strategy often used in social and business research to quantitatively analyze large text corpora [70]. Initially, we coded the twenty-one papers Sedlmair et al. used to derive the conceptual VPSA framework [9]. After that, we extended this literature corpus of practical VPSA applications using a four-step strategy. First, we searched research databases using the query "visual parameter space analysis." Second, we looked for papers that directly referenced the work by Sedlmair et al. [9]. Third, we examined the references of all papers found in the first two steps. Lastly, we searched for papers that referenced works in our current pool but did not cite the work by Sedlmair et al. In this manner, we identified twenty-four supplementary papers that fulfilled our selection criteria, resulting in a cumulative total of forty-five papers.

Our analysis revealed the need to categorize the applications described in those papers according to the following

aspects: the *tasks* an application aimed to address, the *sampling strategy* employed, the visualizations incorporated in an application, the number of *dimensions* encoded by each visualization, the *encoding channels* utilized in a specific visualization, and whether a visualization represented *inputs*, *outputs*, or *both*. At least two authors coded each paper, and the findings were discussed with the entire group. The supplemental material contains the list of forty-five papers and the populated coding table, as well as a discussion of how our approach aligns with Furniss's [71] five stages of Grounded Theory.

While the applications discussed in those papers may be domain-agnostic in several cases, the main application examples presented can be classified into the following categories: Engineering & Material Sciences (16/45), Arts & Design (10/45), and Biology & Medical Domain (10/45). The remaining examples fall into the categories of Operations Research (2/45), Climate & Geo-Science (3/45), and Machine Learning (2/45). Two analyzed applications dealt with generic datasets and could not be classified accordingly.

3.2 Requirement Analysis (L2)

In subsection 2.2), we identified *Cost* and *Trust* as two pivotal factors for the successful adoption of the proposed system by domain experts. They form the basis for our **Key Goals**, namely that the system must impose *Low Cost (Key-I)* on the user regarding time and cognitive effort, and the user must gain *High Trust (Key-II)* in the system and the general VPSA method. In order to achieve those key goals, we derive a set of *Design Goals* and *Requirements*.

Design Goals: Aligning with the design guidelines for domain experts by Wong et al. [30], we determined four design goals. The visualization design space must be *simple yet expressive (DG-1)* in order to solve high-level problems but not overwhelm visualization novices. A strategy to lower Cost and increase Trust is to *activate domain knowledge (DG-2)* which, for instance, can be achieved by including visualization options a domain expert most likely knows and showing them with familiar data. In order to improve Cost and Trust further, the system has to *promote learning (DG-3)*, e.g., by including explanations and providing means so users can transfer existing visualization knowledge to more advanced visualization techniques. Last but not least, a crucial factor in gaining Trust is to *provide transparency (DG-4)*, especially in terms of why a specific visualization was recommended to achieve a particular high-level task.

System Requirements: in order to support those Design Goals, we derived several requirements from the individual requirement analysis outlined in the papers of the meta design study. One requirement is that the system has to be *easy to set up (Req-1)* in terms of installation and initial data preparation. Another requirement we often found was that some kind of *overview (Req-2)* is necessary to understand the analyzed model better. Further, an application must provide means to *navigate a multi-dimensional parameter space (Req-3)* to explore the input-output relations of the model interactively. The remaining generalizable requirements can be described as a variation or a detailed description to support one or several of the *analysis tasks (Req-4)* outlined in Table 1.

	Scatterplot	SP
	(weighted) Density Contourplot	wDCP
MDMV	Scatterplot Matrix	SPLOM
	reduced SPLOM	rSPLOM
	Point Scales	PSc
	Parallel Coordinates	PC
	Histogram	Hist

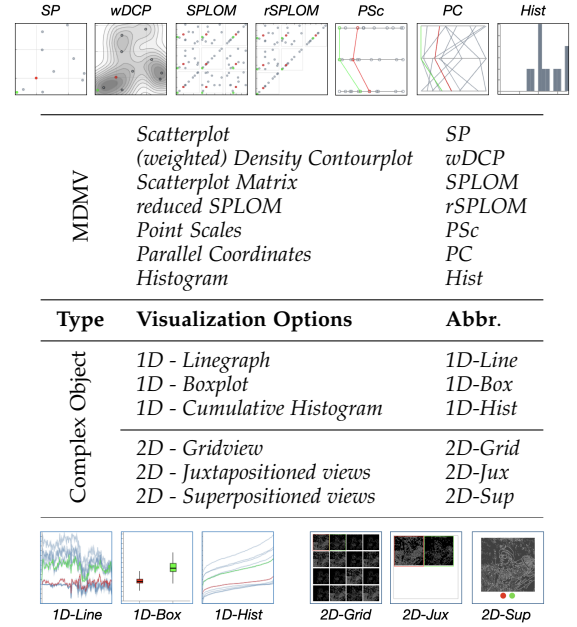| **Type**	**Visualization Options**	**Abbr.**
	1D - Linegraph	1D-Line
	1D - Boxplot	1D-Box
Complex Object	1D - Cumulative Histogram	1D-Hist
	2D - Gridview	2D-Grid
	2D - Juxtapositioned views	2D-Jux
	2D - Superpositioned views	2D-Sup

Fig. 3: Overview of the Visualization Design Space, split into multi-dimensional/multi-variate (MDMV) visualizations, and complex object visualization.

Constraints: RSVP will provide *no sampling support (Con-1)*. We assume that users are capable of sampling their model, at least by using random or Cartesian grid sampling. Most applications we researched focused on visualization solutions that provided multiple interconnected views to study a few hundred runs. For that reason, the tool should be capable of providing solutions for *no more than a couple of hundred runs (Con-2)*. For a single run, some models can output a series of complex objects, e.g., time-varied images. We focus on visualizing single complex objects per run, i.e. *no series of complex objects (Con-3)* need to be supported. Further, integrating models for direct re-sampling (also known as integrated sampling [9]) is an often complicated and error-prone procedure. Therefore, the tool will offer *no direct model interface (Con-4)*.

3.3 Visualization Design Space (L3)

We derived a common design space meeting the requirements of *DG-1* by reviewing the findings from the content coding stage and extracting visualizations that appeared in at least two different applications. An overview of the visualization options in the resulting design space is presented in Figure 3. However, not every visualization fitting the previous criterion ended up in the final design space. For instance, we found four occurrences of Heatmaps, three Density Plots, and two Contour Line Plots. We combined those into *(weighted) Density Contour Plots (wDCP)* since Heatmaps would have required a dense grid sampling strategy which we wanted to avoid (Con-1). As previously outlined, data analyzed in VPSA typically consists of *scalar parameter values* and often *complex objects*, which cannot be described with a single variable without losing information. Therefore, the design space is conceptually split into *multi-dimensional, multi-variate (MDMV)* visualizations and *complex objects* visualization options. The final design space includes three 1D and three 2D complex object visualizations.

App	Hist	SP	SPLOM	rSPLOM	PSc	PC	wDCP	CO	Task	Sampl
Filter:								T_opt stoch		
#1	S:5 S:4			S:5	S:5 S:4				T_{opt} T_{out} T_{part}	reg
#2			S:3+C+B S:4+C+B					1D-Box 2D-Jux	T_{fit} T_{unc}	stoch
#3	S:6 S:2	S:8 (1)				S:8		2D-Grid 2D-Sup	T_{opt} T_{sens} T_{part}	stoch

Fig. 4: A visually enhanced representation of the coding table. For the current filter criteria T_{opt} and stoch, applications #1 and #2 are filtered out. Markers in MDMV visualization columns (white headers) contain information about how many dimensions were encoded (S)patially, if they encoded (C)olor or (B)rightness, and if they encoded inputs, direct outputs, or derived outputs.

It is noteworthy that several reviewed applications included a detailed view of a 3D object. However, these views were typically associated with a unique research contribution, thus making them uncommon by definition.

3.4 VisRec Strategy (L4)

The coding table described in subsection 3.1 provides the basis for our proposed VisRec strategy. Figure 4 shows a visually enhanced example of that table. Each row represents an application and columns represent visualization options (MDMV and complex objects), tasks, and the underlying sampling strategy used to generate the runs. The latter two form an *evaluation space* [72] over the visualization design space. This setup enables filtering for applications that address specific tasks and discovering the visualizations and settings they employ to accomplish their objectives. A rigorous analysis of the papers in the meta design study and the subsequent content coding enabled us to derive the rules for the VisRec strategy summarized in Table 2a. Our findings indicate that each task has a unique and fundamental problem-solving approach, regardless of the underlying data type. In the following paragraphs, we will present these approaches and provide further insights into our research.

Achieving *optimization* (T_{opt}) requires providing the best possible overview of objects and parameters. Therefore, without an objective function, the requirement for overview (Req-2) is not just a general feature but a necessity for optimization. In the context of MDMV, providing an overview means offering as much information about the data as possible while considering visual clutter, data occlusion, and cognitive overload. Based on the Mackinlay criteria [35], we refer to the strategy of presenting spatial information clearly and concisely as "spatial expressivity" (see Table 2b for a summary of its high-level rules). Providing an overview of complex objects is typically accomplished through overplotted function graphs (1D) or presenting options in a grid layout (2D). Optimization often forms the basis for other, more specific tasks, which extend T_{opt}'s overview with additional visual features or views.

Affiliating solutions is the primary approach for *fitting* tasks (T_{fit}). The color channel is often used to represent the output of an objective function to establish connections between different MDMV visualizations. When dealing with 2D objects, we discerned a preference for juxtaposed comparison views.

When *uncertainty* (T_{unc}) was quantified by a numerical parameter, the color channel was used to display it, preferably using black or white at opposite ends of the spectrum [15], [73]. If the color channel was already used for other purposes, the opacity channel was used instead [74], [75]. If the underlying distribution was available as a 1D object, selected items were displayed as Boxplots [76], [77]. Both T_{fit} and T_{unc} worked typically on top of visualizations suggested by T_{opt}.

To visually distinguish *outliers* (T_{out}) from other data points, visualizations often used the smallest available marks for a given data type. This technique helped separate outliers from the rest of the data. Six applications out of eight supporting this task used Scatterplots or Scatterplot Matrices (0D-mark) for MDMV visualizations (the remaining two [1], [78] using bespoke visualizations), and four applications out of five encoding 1D objects used overplotted Linegraphs (1D-mark).

In *sensitivity* tasks (T_{sens}), the objective is to identify parameter settings with either low or high impact on model outputs. Line-based visualizations help identify converging (dense) or diverging (sparse) sections, which makes Density Contour Plots and Parallel Coordinates the recommended choices for MDMV visualizations. Linegraphs are effective for identifying local sensitivities in 1D objects, while Cumulative Histograms can help in finding global thresholds. Superpositioned views support determining sensitive regions for selected 2D objects.

The *partitioning* task (T_{part}) tries to identify different types of model behavior, especially when there is no objective function to guide the analysis. The key is to find boundaries that separate groups that behave similarly. His-

TABLE 2: Overview of the VisRec-strategy for the VPSA design space

Task	Strategy		MDMV	Complex Object
T_{opt}	Overview	Channel-based	Spatial expressivity	1D-Line, 2D-Grid
T_{fit}	Affiliation		Overview + Color	2D-Jux
T_{unc}	Attenuation		Overview + Brightness	1D-Box
T_{out}	Separation	Mark-based	Point-based (0D-mark)	(1D-Line)
T_{sens}	Con-/Divergence		Line-based (1D-mark)	1D-Hist, 2D-Sup
T_{part}	Summarization		Area-based (2D-mark)	2D-Grid $^{(-)}$

(a) high-level overview of the task-oriented VisRec strategy

Dims#	In_{CP}-Reg	In_{CP}-Stoch; $Out_{dir,der}$
(1)	(PSc+Hist)	(PSc)
2	wDCP+Hist	SP
3	SPLOM+wDCP+Hist	SPLOM
4	PC+wDCP+Hist	(r)SPLOM
5	PC+wDCP+Hist	rSPLOM
6	PC+Hist	rSPLOM
7-9	PC	PC
10+	(PC)	PSc

(b) Spatial expressivity

tograms are commonly used because they abstract away details and help to focus on the bigger picture. When analyzing 2D objects, a Grid Layout showing elements within a selected range but hiding filtered ones can achieve a similar effect while still providing detailed information about the individual items.

From a high-level perspective, our findings suggest that tasks T_{opt}, T_{fit}, and T_{unc} involve encoding specific *channels*, whereas addressing T_{out}, T_{sens}, and T_{part} requires selecting appropriate *marks*. Within the context of our research, the specific marks and channels were even distinct for the various analysis tasks (see MDMV-column in Table 2a).

4　RSVP System

The *Rapid Suggestive Visualization Prototyping* (RSVP) system is a practical implementation of the findings from the meta design study. In this section, we present a walk-through of the RSVP system, provide some technical details, and offer a qualitative result inspection about how the system aligns with previously outlined design goals and requirements.

4.1　Walk-through

RSVP enables users without specific visualizations or programming skills to design dashboards tailored to their specific data and needs regarding VPSA. The user interface tightly integrates an overview of available visualization options with an unobtrusive recommendation strategy to foster transparency and learnability. RSVP consists of four main components, as outlined in Figure 5: the *Data-Selection Panel* (DSP), the *Overview Area* (OVA), the *VisRec Interface* (VRI), and the *Visualization Dashboard* (VDB). These components are shown in Figure 5 and will be described in detail in the following sections.

DATA-SELECTION PANEL (DSP) : Data gets loaded in a simple CSV format directly into the textfield in the data panel (DP). The header must contain the names of the dimensions, and rows contain the details about the individual runs, namely the parameter settings and the results. Once loaded, data dimensions are represented in the form of draggable markers (see Figure 6). The color of a marker represents the underlying data type. RSVP detects three
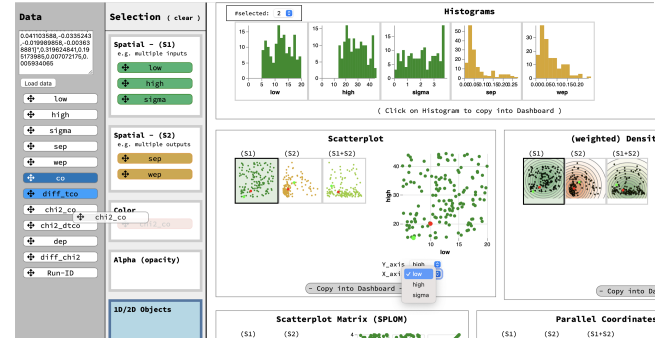

Fig. 6: The Data-Selection Panel (DSP) and parts of the Overview Area (OVA). Dimension is actively moved into the Color field. The interactive switch for the Scatterplots x-axis shows the three available dimensions from Spatial field (S1).

different data types: numerical parameters (scalar values, white), 1D objects (arrays, blue), and 2D objects (links to image files, dark blue).

The markers can be dragged and dropped into channel fields in the adjacent selection panel. Available channels for direct data encoding are spatial, color, and opacity (see subsection 3.3). RSVP offers two spatial channel fields, S1 and S2, so MDMV visualizations split along inputs and outputs
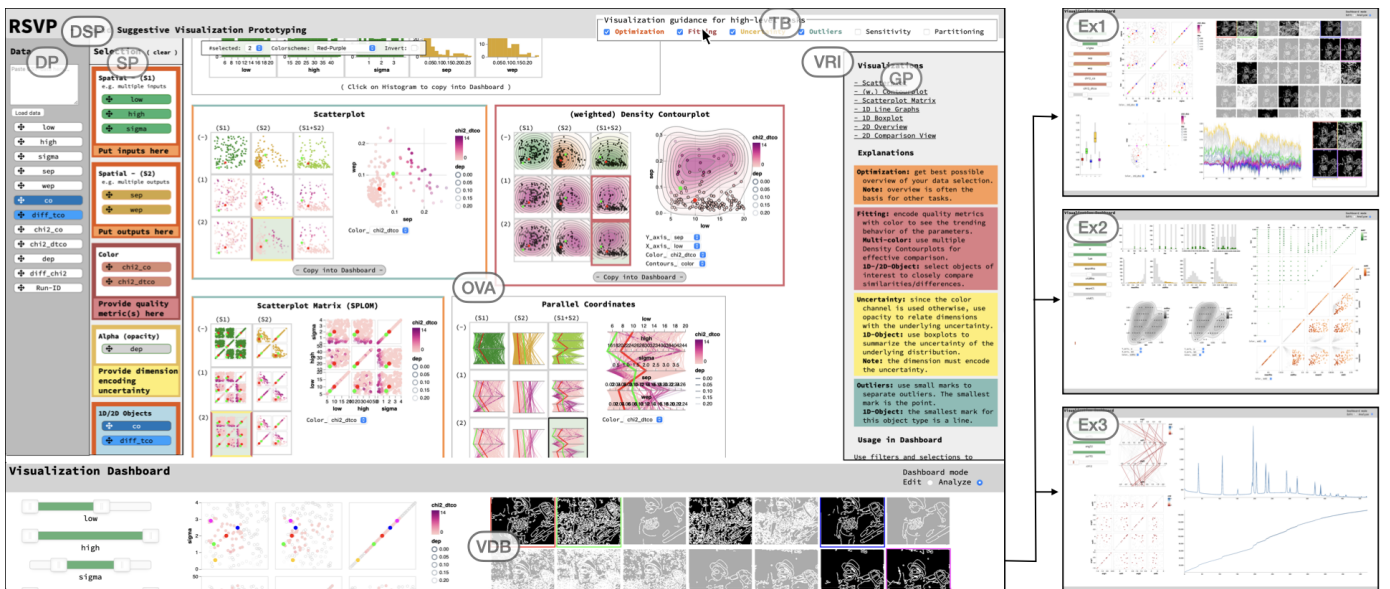

Fig. 5: **Left:** Overview of RSVP system for VPSA. *System left*: Data-Selection Panel (DSP). Dimensional markers in the data panel (DP) represent the user-provided CSV data. Those markers can be dragged into the channel fields in the selection panel (SP), where white MDMV markers change their underlying colors. *System center*: the Overview Area (OVA) displays all available visualizations from the design space with user-selected data. The colors used in the visualizations match the marker colors in the selection panel. *System top-right*: the VisRec Interface (VRI) is comprised of several components. Selecting tasks in the taskbar (TB) opens the guidance panel (GP) and injects recommendations directly into the overview area (OVA) and the selection panel (SP) in the form of colored frames. *System bottom:* the Visualization Dashboard (VDB) enables users to perform interactive data analysis. Visualizations can be copied from the overview area into the dashboard.
Right: Dashboard instances (Ex1) - (Ex3) showcase examples created from different data sets in RSVP within minutes.

can be displayed simultaneously in the overview area using small multiples displays (SMDs). MDMV markers, when placed into channel fields, adjust their color to correspond with their appearance in the overview area, enhancing orientation. Complex object markers must be positioned within the object field to be visible in the overview area.

Please note that the colors utilized in Figures 2 and 4 are selected to align with the colors employed in the RSVP system for markers in channel fields Spatial S1, Spatial S2, and Color. These fields are frequently used to represent inputs, direct outputs, and derived outputs, respectively.

OVERVIEW AREA (OVA) : displays all visualizations from the design space (see Figure 3) in a 2-dimensional layout, directly encoding user-provided data from the selection panel (SP). The top area contains MDMV visualizations, and the bottom displays complex object visualizations. With the exception of Histograms, MDMV visualization options consist of a Small Multiple Display (SMD) and a detail view [63]. Selecting a small multiple will load the according data settings in the detail view for this visualization. The SMDs show subsets and combinations of user-selected dimensions that could not be presented simultaneously otherwise. The layout of the SMDs tries to follow the same rules for all the visualization options incorporating it, depicted in Figure 7. Each view in a particular column encodes the same spatial dimensions. The first column encodes dimensions
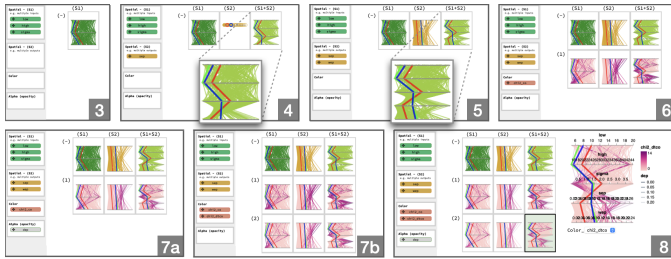

Fig. 7: The SMD behavior using Parallel Coordinates as an example. The dark boxes indicate the numbers of encoded dimensions. Configurations 4 and 5 highlight the differences for the (S1+S2)-columns. Configs 7a and 7b encode seven dimensions slightly differently, resulting in additional rows. Configuration 8 is displayed with the according detail view for the currently selected small multiple.

from the first spatial field (S1), and the second column from the second spatial field (S2). The third column encodes the combination of both spatial fields (S1+S2) (see the highlights in Figure 7 configurations 4 and 5 for an example). Our literature analysis did not reveal an instance where inputs and outputs were combined into a single *(r)SPLOM*. Therefore, *SPLOM* and *rSPLOM* do not provide a third column to avoid unnecessary rendering overhead. The first row in the SMD encodes spatial fields only, where the various small multiples are displayed with categorical colors that link them visually to the underlying spatial fields in the selection panel. Additional rows encode the selections for the color and opacity channels. If either channel gets encoded more than once, multiple additional rows will be displayed (Figure 7, configurations 6 through 8 show this particular behavior for different encodings).

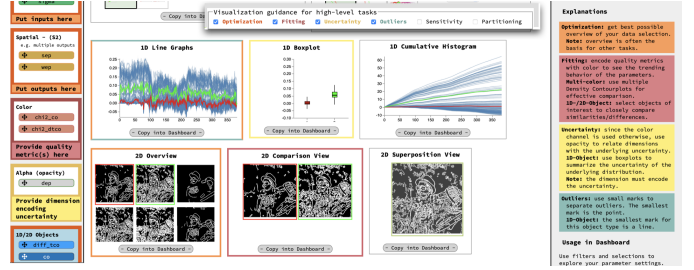If a visualization can only display some of the

Fig. 8: Complementary view showing the complex objects for the recommendations in Figure 5.

dimensions from the selection panel, interactive switches will appear so the user can adapt them during analysis. Examples of such situations include cases where *SP* or *wDCP* have to encode three or more spatial dimensions (see Figure 6), or any visualization has to encode more than one dimension with the color or opacity channel (see Figure 9). A panel at the top of (OVA) allows for global adjustments of the visualizations, like the number of pre-selected data points in the overview for comparison purposes or the color scheme for dimensions encoded in the color channel.

VISREC INTERFACE (VRI) : consists of the *taskbar* (TB) and the *guidance panel* (GP), and spans over the *overview area* (OVA) and the *selection panel* (SP). RSVP can provide recommendations for up to four tasks simultaneously. When seeking VisRec guidance, T_{opt} is automatically selected, as it typically serves as the foundation for other tasks. Each task is associated with a unique categorical color (see Table 1), which makes it possible to track the recommendations for a specific task across the different sections.

The taskbar (TB) in the top right shows the tasks for which users can seek recommendations. Hovering over a specific task provides a brief explanation about it. Selecting one of the tasks triggers the recommendation process. Recommendations are provided as colored frames around the visualization types and single instances in the small multiples in the overview area (OVA). For tasks optimization T_{opt}, fitting T_{fit}, and uncertainty T_{unc}, dimensions have to be encoded according to the channels supporting the respective tasks (see Table 2a). RSVP frames the channel fields in the selection panel (SP) with the respective task color and provides context information on which kind of dimensions are supposed to be encoded (*inputs*, *outputs*, *derived* values and values *quantifying uncertainty*).

The Guidance Panel (GP) provides helpful additional information regarding the recommendations. It consists of three parts. On top is a list of the recommended visualization types. This list links to the respective options in the Overview Area (OVA). The middle section contains explanations and background information on how recommended visualizations are supposed to support respective tasks. At the bottom are hints on how to interact with the visualization dashboard to solve these tasks.

Figures 5 and 8 show an example recommendation for tasks T_{opt}, T_{fit}, T_{unc}, and T_{out}. Further examples for recommendations are provided in subsection 5.2, the supplemental material, and the accompanying video.

VISUALIZATION DASHBOARD (VDB) : visualizations

can be copied from the Overview Area (OVA) into the Visualization Dashboard (VDB). The dashboard allows the combination of an arbitrary number of visualizations to perform multi-view analysis. Individual visualizations can even be combined into more complex ones. Figure 9 shows an example where Point Scales, Histograms, and interactive sliders are combined into parallel histograms. A slider appears for each dimension loaded into the dashboard, providing a consistent way of filtering data points across all visualizations. The color of the slider matches the color of the dimension in the selection panel. Selecting a single data point within a visualization will also highlight the same element in all the other visualizations in the dashboard. The dashboard offers
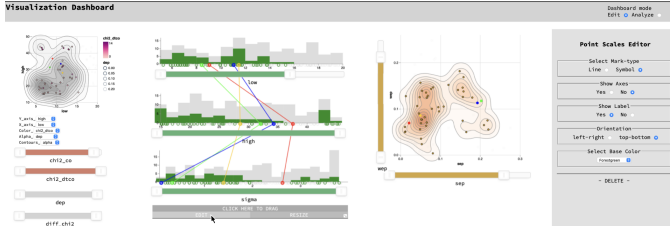

Fig. 9: Visualization Dashboard (VDB) example in *edit*-mode with the single-view editor (right) opened for the *Point Scales* used in the parallel histograms (green).

two modes, *edit* and *analyze*. In *edit*-mode, hovering over a visualization or slider reveals a context menu that allows for manipulating this particular element, like moving, resizing, or changing its attributes. The attributes can be edited via the single-view editor. When in *analyze mode*, dashboard design and attributes can not be changed, and the user can focus on analyzing the dataset.

4.2 Technical Aspects

RSVP is a single-page web application built with standard web technologies (HTML, CSS, JavaScript). It operates without a backend; all computations occur within the browser. The application leverages several helper libraries for various tasks, including parsing CSV data[1], rendering HTML templates[2], enabling drag-and-drop functionality[3] in the data selection panel (DSP), and implementing dragging and resizing[4] as well as the data sliders[5] in the visualization dashboard (VDB). MDMV visualizations and 1D complex object visualizations are generated using the Vega visualization toolkit [79], while 2D complex objects are implemented using plain HTML and CSS. The recommendation algorithm is designed as a series of cascading rules, following a similar approach outlined in BOZ [41]. Further technical details about the implementation of the VisRec strategy can be found in the supplemental material.

4.3 Qualitative Result Inspection (QRI)

In this section, we provide the rationales of how our proposed solution fulfills the requirements and design goals

1. https://www.papaparse.com
2. https://underscorejs.org
3. https://bevacqua.github.io/dragula/
4. https://interactjs.io
5. https://refreshless.com/nouislider/

outlined in subsection 3.2. The two Key Goals *Low Cost (Key-I)* and *High Trust (Key-II)* will be discussed in more detail in subsection 6.1.

Easy setup (Req-1): RSVP is a web application that can be used without an account. It requires a simple CSV file as input, which can be created with any modern spreadsheet software. Further, the system does not require a specific sampling strategy, but the user can choose the most feasible one. In the dashboard (VDB), visualizations and sliders for filtering can be arranged and resized via drag-and-drop. All these aspects ensure an easy setup for the user.

Overview (Req-2): RSVP provides an overview of basically all elements associated with user interaction. It shows the user-provided data dimensions in the data panel (DP) and all encodable channels in the selection panel (SP). The taskbar (TB) presents the available analysis tasks, and the overview area (OVA) shows all the available visualizations directly encoding data . This enables a quick overview of the parameters and complex objects from many different perspectives. Visualizations offered by RSVP are extracted from applications mainly featuring *global-to-local* navigation strategies. Hence, visualization dashboards (VDB) composed of these visualization options feature an initial overview of the data by design.

Navigating Parameter Space (Req-3): Available operations for navigation during an analytical session in the visualization dashboard (VDB) are filtering, zoom, and selection of individual runs, therefore following Shneiderman's information-seeking mantra ("Overview first, zoom and filter, then details-on-demand") [80]. To navigate the design space, the system presents MDMV visualizations in small multiples and large singles, akin to the design of van den Elzen and van Wijk [63] for visualization exploration. RSVP supports the user in navigating the vast design space by suggesting appropriate visualizations to solve user-defined high-level tasks. This support also includes suggestions for navigating the parameter space by indicating to the user how to encode the various dimensions in the parameter space for visual exploration.

Analytical Task Support (Req-4): RSVP supports domain scientists in solving analytical tasks by indicating which visualizations to choose and how to encode them. It further informs the users how to interact within the dashboard to achieve those tasks.

Simple yet expressive (DG-1): Our approach focused on creating a design space that balances simplicity for novice users with the expressiveness needed for complex analytical tasks. Instead of favoring complicated special purpose views, our approach focuses on finding common and likely easier-to-understand visualization options, which can further be composed into complex multi-view visualizations, supporting specific, high-level tasks.

Activate Domain Knowledge (DG-2): Practical recommendations for activating domain knowledge are using terminology and visualizations familiar to users [30]. Since RSVP is domain-agnostic, it avoids using domain-specific terminology in labels or explanations. However, users are likely familiar with the terminology used in the data they provide. Moreover, the dimensions and data points are likely meaningful to them. The key idea is that the data will initially help improve understanding of the various
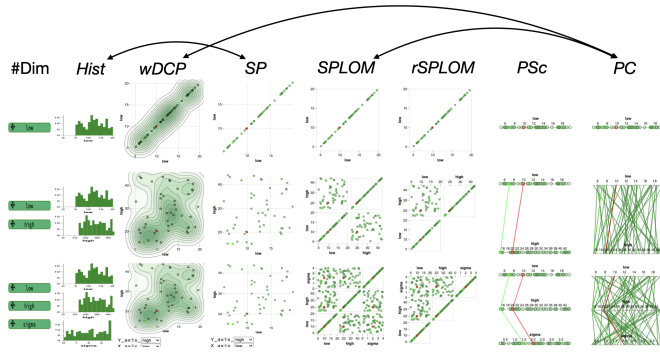
Fig. 10: RSVP promotes learning by adjacency and progressive reveal. Columns show which visualizations are adjacent in the Visualization Overview (OVA) (the arrows on top show further adjacencies in the 2-dimensional layout) and how they are supposed to show similarities and differences between adjacent visualizations. Rows show how the complexity of visualizations is progressively revealed when adding more dimensions.

visualizations. RSVP presents all available visualizations immediately encoded with the user-provided data. This might not only improve understanding but seeing the data through different lenses of multiple visualizations might provide immediate insights or raise exciting questions [62], [81]. Since several provided visualizations (like scatterplots, histograms, and line charts) are the subject of teaching within the K-12 curriculum [82], users are likely familiar with at least some of them.

Promote Learning (DG-3): RSVP promotes learning about new visualizations by positioning similar visualization options closer together in the overview area and encoding them directly with user-provided data. The idea is to make it easier for users to spot and compare similarities and differences between known and previously unknown visualizations. It should also improve the understanding of familiar visualizations in the context of the current problem. Furthermore, to make learning about visualizations even more accessible, the complexity of each visualization is initially reduced and then revealed progressively [29], [30] when additional dimensions get added. As presented in Figure 10, when only a single dimension is encoded (visualizations requiring at least two spatial dimensions encode the same dimension on both axes), several MDMV-visualization options look very similar. Adding a second dimension shows how previously similar visualizations begin to differ from each other. Adding a third dimension further distinguishes visualizations capable of spatially encoding more than two dimensions (like SPLOM or PC). After each step, the users can familiarize themselves with the more complicated visualizations. To avoid misunderstandings caused by non-self-descriptive tasks, our system explains each task when hovering over them in the taskbar. Additionally, RSVP explains to users why recommended visualizations are supposed to support specific tasks.

Provide Transparency (DG-4): These explanations are not only helpful for learning but additionally provide transparency and should ultimately support building trust in the system [33]. Recommendations are presented in a non-obtrusive way. RSVP presents all available visualizations for a given datatype and highlights recommended ones (a guidance strategy known as "orientation") [83]. This approach

allows users to draw conclusions about recommended visualizations and make alternative decisions if necessary.

5 EVALUATION

The three most prevalent evaluation strategies we found in the meta design study, reflecting general evaluation trends in the visualization community [84], are *case studies*, *usability studies*, and *qualitative result inspections (QRI)* [85]. The latter we already discussed in subsection 4.3. In this section, we will present the results of a usability study and two real-world case studies from different problem domains utilizing RSVP. Additionally, the supplemental material presents and discusses another case study that was conducted using an earlier prototype of RSVP.

5.1 Usability Study: Edge Detection Algorithm

For this study, we recruited four undergraduate students majoring in media informatics and two graduate students specializing in computer graphics. Our choice for the number of study participants is based on recommendations for determining the sample size for usability tests [86], [87]. The recruits, four males and two females, were between 21 and 36 years old ($\mu = 25.83, \sigma = 5.01$). The rationale for recruiting these users stems from the potential benefits they could gain from using a tool like RSVP. They frequently interact with diverse computational models, where understanding the connections between inputs and outputs is crucial for fostering learning and comprehension. However, such problems do not warrant the collaboration with a visualization expert in order to perform a design study. Participants had to analyze data sets generated from the edge detection algorithm used in the running example, utilizing the same generic image in order to avoid domain complexity. Nonetheless, the problem of creating a contour outline of an image finds application in many different areas (see section 2).

Sessions were conducted online via Zoom using a hosted version of RSVP. Each session took 60 minutes. The participants received a five-minute introduction to VPSA and a five-minute tutorial on how to use RSVP. Afterward, they had ten minutes to explore RSVP's various functions and ask questions. The primary analysis session took thirty minutes, during which participants had to create dashboards to analyze three increasingly complex data sets. Each dataset was introduced briefly at the start of each iteration and came with a challenge that participants needed to address within ten minutes. The challenges were (1) identifying optimal parameterizations for the data set, (2) determining sensitive areas and which dimensions influenced them, and (3) comparing and evaluating the quality of different objective functions. Figure 5 - (Ex1) shows an example dashboard created during an analysis session. The nature of the questions was open-ended, which means there were no right or wrong answers. Instead, at the end of each iteration, the users had to rate their confidence in their findings on a five-point Likert scale (see Table 3), which is a common evaluation approach in such open-ended settings [88].

During the final interview, we asked the participants to fill out a *System Usability Scale* (SUS) survey [89], a well-known and widely adopted approach to measuring the

TABLE 3: Confidence ratings of the study participants, ranging from *not confident at all* (1) to *very confident* (5)

	P1	P2	P3	P4	P5	P6	∅
Challenge 1	4	4	5	5	2	4	4
Challenge 2	5	4	3	4	3	5	4
Challenge 3	5	4	5	5	5	5	4.833
	4.67	**4**	**4.33**	**4.67**	**3.67**	**4.67**	**4.277**

usability of a system. An abbreviated version of the questions, along with the answers given by the participants, are presented in Figure 11. The complete results are available in the supplemental material. Converting the users' answers into the final scores yields results between 70 and 92.5 points, with a mean of 82.5 points. According to Brooke, average scores above 80 are considered *very good* [89].
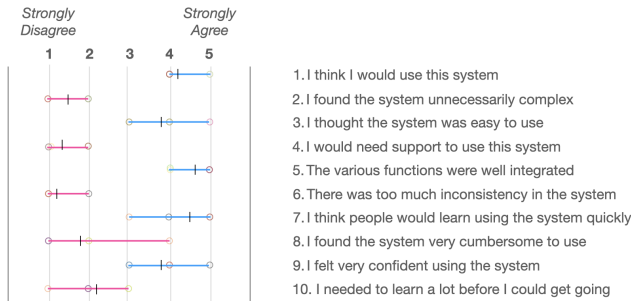

Fig. 11: Results for the SUS survey, showing range (colored lines) and means (vertical black bars) of given answers. Positive questions (odd-numbered, blue lines) aim for a high score, and negative questions (even-numbered, red lines) aim for a low score.

5.2 Case Study I: Crystal Powder Diffraction

Our first case-study user is a physics professor who specializes in solid-state materials. He studies crystalline structures using a procedure known as *powder diffraction* and compares the results to simulated model data using a χ^2-metric to learn about molecular behavior inside the crystal. The knowledge gained from such experiments helps the physicist to grow new crystals with desired features.

The physicist provided fifty random samples for four parameters of interest and 1D outputs and derived χ^2-metrics. Due to timing constraints, the analysis was divided into two sessions: one in-person and one online. At the beginning of the in-person session, he received a brief introduction
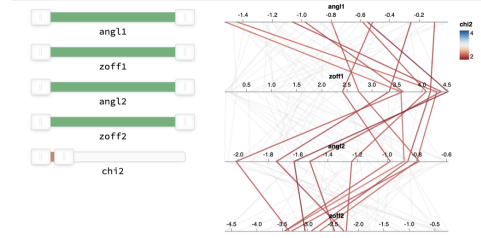

Fig. 13: Detail view of the filters and Parallel Coordinates from (Ex3).

to RSVP, followed by time to experiment with the system and ask questions. The recommendations that underpin the dashboard depicted in Figure 5-(Ex3), which showcases the most noteworthy insights, are shown in Figure 12.

The Parallel Coordinates show that parameter settings to obtain a low χ^2 result scatter over the entire range for parameters *angl1* and *angl2*, but they become narrower for *zoff1* and *zoff2*. This behavior indicates that *angl1* and *angl2* have little to no influence to get a good χ^2 result, whereas *zoff1* and *zoff2* do. The physicist was aware of this behavior but was surprised that such a low-effort approach using a coarse sampling strategy combined with a relatively simple visualization could reveal what he had learned from long computational cycles and multiple sequential experiments.

Studying the complex object visualizations revealed insights previously unknown to the scientist. Generally speaking, the physicist tries to figure out how the molecules fit together inside the crystal. The curves in the 1D-Linegraph represent crystal structures in reciprocal space, a Fourier transform of the real space [90]. The peak locations and spacings represent the periodicity of the crystal in three space dimensions, and peak heights indicate details about the molecular structure. Usually, the domain scientist studies these curves one at a time instead of overplotting the results of several experiments in a single graph.

As presented in Figure 14, the four input parameters highly influence the peak at position 196, a well-known circumstance to the scientist. However, he was surprised by the high variance of suitable candidates for good χ^2 results (see Figure 14- Z1). He expected the margin of error at this position to be way lower for suitable candidates. Another finding was that the peak at position 291 is invariant under
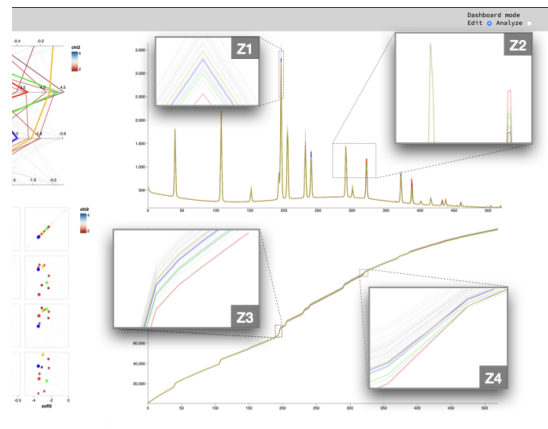

Fig. 12: Recommendations for tasks optimization T_{opt}, fitting T_{fit}, and sensitivity T_{sens} for the Powder-dataset.

Fig. 14: Showing the 1D-Linegraph and the 1D-Cumulative Histogram from dashboard (Ex3), with several good candidates selected and presenting zooms for multiple areas of interest.

these four parameters, meaning it remains static for any parameterization (see $\boxed{Z2}$, left peak). It will require further research on why the molecules are invariant to depth (zoff) and rotation (angl) at this particular frequency. Combining these findings and studying them within the *1D Cumulative Histogram* view revealed further insights. Highlighting runs with good χ^2 results show that they do not change their positions relative to each other between the peaks at locations 196 and 291 (see $\boxed{Z3}$ and $\boxed{Z4}$, respectively). In other words, the area between these two peaks is stable regarding the objective function. From a global perspective, the peak with the highest variability is followed by a relatively stable area, culminating in a perfectly stable state at position 291 before massive changes set in again immediately after this location (the right side of $\boxed{Z4}$ shows how the highlighted runs begin crossing each other). The scientist was completely unaware of this global pattern that actually challenged his understanding of this crystal entirely.

5.3 Case Study II: Lighting Design Optimization

Our second domain scientist is a researcher who aims to automate the placement of light sources in official buildings by adapting the lighting configuration through a gradient-based optimization approach. Office environments and official buildings must meet specific lighting standards for different areas, like desk surfaces. Current commercial lighting design tools necessitate manual adjustments of luminaires to reach the desired outcome. This new approach would allow designers to directly set target radiances for surfaces, eliminating the need to manually adjust the luminaires to achieve the intended results.

The researcher was intrigued by the general VPSA approach and its potential benefits to his work. He conducted 685 runs of a two-level office room with four lamps. The positioning of the lamps was unconstrained, allowing them to be placed anywhere in the room. Each lamp's position in 3D space constituted twelve parameters, along with the objective function results for each configuration. Additionally, he provided gradients for the optimization step, including the gradient norm, adding thirteen more parameters, for a total of twenty-six to analyze. Generating the samples and converting them into the required format took approximately forty-five minutes.

The analysis session was conducted in person on a 2021 16-inch MacBook Pro equipped with an Apple M1 Max chip and 64GB of memory, running a locally hosted version of RSVP. After receiving a brief introduction to RSVP, the researcher began with the analysis, during which he could ask questions about the system at any time. He experimented with combining the dimensions in different ways and studied the recommendations and what they meant. During the session, he created numerous dashboards and had several interesting insights. The dashboard presented in Figure 15 summarizes some of the most interesting ones. This is the outcome of RSVP's recommendation for task optimization when encoding all parameters (PSc for both inputs and outputs) and subsequently adding the recommendation when only including the y-dimensions of the lights (SPLOM).
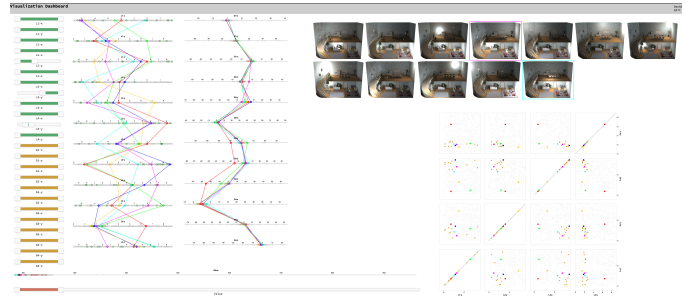

Fig. 15: Dashboard created during the Case Study on Automated Lighting Optimization

In this dashboard, the objective function value is filtered to a relatively low level, highlighting options with favorable values. An interesting observation emerged regarding the y-positions of the lamps, indicating a necessity for at least two lamps on the bottom level and one on the top level of the office, while the position of the remaining lamp could be more flexible. The researcher could deduct this from the images and confirm it with the recommended plots. Although not surprising, since effective lighting is mainly required for three surfaces in this particular room, this insight prompted the realization that a simple sampling strategy could address a challenge in their algorithm: determining the minimum number of lamps to place in a room initially Apparently, a straightforward random sampling strategy could be used with varying numbers of lights, and those yielding feasible objective values could then serve as the starting points for their optimization algorithm.

The other insight pertained more to the system itself. While all values are below a required threshold, examining the rendered images reveals that not all of them are visually pleasing. Several results display bright spots on the wall (see Figure 16a), which should be avoided in efficient lighting design. On the other, elements that should receive at least some light can appear rather dark, like the staircase in Figure 16b. Both issues can be addressed by adjusting the illumination constraints on these objects. While relatively easy to achieve in theory, it is often overlooked in practice to add all constraints that are not part of the requirement specification. RSVP could be utilized for debugging these configurations. Rendering a second image from a different perspective would essentially provide a 360° view of the office, allowing one to scrutinize good results for possible mistakes stemming from missing negative constraints.
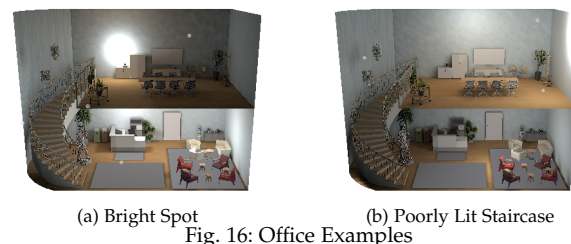
(a) Bright Spot   (b) Poorly Lit Staircase
Fig. 16: Office Examples

6 DISCUSSION

In this section, we present our findings regarding the evaluations and our two key design goals, followed by additional

analysis concerning the design aspects of RSVP. We conclude this section by outlining the limitations and potential directions for future research.

6.1 Findings

Our basic assumption for the meta design study proved to be correct. It was possible to extract an expressive visualization design space and a task-oriented visualization recommendation strategy from existing literature, utilizing insights from the conceptual VPSA framework. In the following, we will analyze how our design achieves our two key goals.

Low Cost (Key-I): The physicist reported that sampling the model and formatting the data took approximately sixty minutes. This estimate includes the time required to figure out how to perform the sampling. He believes this process could be reduced to a few minutes with some exercise.

Five out of six participants of the usablity study were unaware of at least one visualization offered by RSVP. However, no participant had trouble interpreting and using any of them. All the users mentioned that it was beneficial to see the visualization options side by side encoded with actual data, which made learning by comparison easy. Three participants noted that the step-by-step introduction was helpful for them. One of them noted: *"Seeing all the visualizations [encoded] with all the dimensions immediately would probably have been overwhelming. But seeing how similar they are in the beginning was very interesting and made it easy to follow as you added new dimensions."*

The domain scientist was pleased to see his data visualized right away. He had prior experience with a design study that took months and several iterations before he saw actual visualizations of his data. *"Back then, I entered the design study with the prospect of learning something about visualizing my data and finding out if it could help me with my research. RSVP did all of that, just really fast."*

Regarding the SUS questionnaire, we argue that *high usability* can be interpreted as *low cost* of using a system. As previously reported, RSVP achieved an average score of 82.5 points, which is considered to be *very good* [89].

We take those findings as positive indicators that RSVP achieves Key-I, imposing *low costs* on the user in terms of time and cognitive effort required for using it.

High Trust (Key-II): The domain scientist admitted that he was initially skeptical about the VPSA approach. He could not imagine that such a coarse sampling could reveal anything about the data that he did not already know. However, fast revelations and confirming existing knowledge about the data helped to build trust and further spurred his curiosity. Consecutive findings and insights led to the decision to adopt RSVP for future research. He issued the following statement: *"Although the instruments for taking physical measurements have improved over the years, the way scientists analyze their data has not changed much. VPSA, in general, could change how scientists study their models, and RSVP, in specific, could probably help them to achieve this."*

Trustworthiness in VisRec may not be captured by simple measures, but Dasgupta et al. suggest that trust can be defined as self-calibrated confidence in the analysis outcome [33]. Therefore, the confidence ratings obtained from the usability study indirectly reflect the reliability of both the VPSA approach and the VisRec strategy. The overall confidence score was 4.17 points, which means that usability testers were *rather confident* in their findings. Interestingly, the challenges with the highest complexity had the highest confidence ratings (4.83 points on average). This finding is similar to observations by Dasgupta et al. [33] where they reported that domain scientists' trust in a visual analysis system increased with the complexity of the analysis tasks. The users explained this behavior by the experience gained during the previous challenges and that they would not have known how to evaluate the objective functions otherwise. We take these findings as proof that RSVP also fulfills Key-II, making users feel *high trust* in both the RSVP system and the VPSA method in general.

6.2 Further Design Analysis

As previously discussed, a meta design study diverges from a classical design study in its method of deriving domain knowledge. While the latter emphasizes user-centric design, the former derives domain knowledge not from direct interaction with individuals but rather from written reports. This is also how assumptions about domain scientists are formed. However, to enhance the overall usability of such a tool or to potentially address the needs of individual scientists, iterative and user-centric design remain essential. Therefore, we will delve into these aspects in this section. Furthermore, existing frameworks like ComVis [65] and Visplore [66] generally allow for the creation of more powerful dashboards compared to those possible with RSVP. However, they are not readily accessible to domain scientists, requiring a visualization expert to design and develop a customized application or dashboard before analysis can begin. In contrast, RSVP allows domain scientists to independently develop their own VPSA dashboards without the assistance of a visualization expert. Tableau, however, does not require programming and can effectively create all the visualizations that RSVP can. This raises the question: what factors would motivate choosing RSVP over Tableau? To address this, we will compare RSVP to Tableau and attempt to answer this question.

Iterative Design: The design study discussed in the supplementary material represents an early application of RSVP, demonstrating its initial functionality. At this stage, RSVP only featured a single spatial encoding field, necessitating the sequential creation of visualizations for inputs and outputs. Recommendations were provided in the form of a ranked list of visualizations deemed effective for specific tasks. While this study provided interesting insights for the user, it showed the necessity of collaboration with visualization experts familiar with RSVP and VPSA for the development of comprehensive dashboards.

This experience prompted a redesign of RSVP, introducing two spatial fields to enable simultaneous input and output dimension encoding. Additionally, it prompted a reassessment of the coding table concerning the Vis Rec strategy, culminating in the strategy detailed in Section 3.3.

Furthermore, prior to conducting the usability study, two pilot studies involving an undergraduate student and a Ph.D. student were conducted, resulting in further usability

enhancements. These enhancements included clarifications on task meanings, explanations within the dashboard, and adjustments to color schemes. While the initial color differentiations between tasks and encoding channels were distinct, efforts were made after the pilot studies to align them more closely, facilitating task-channel associations. Moreover, the color scheme for the third column in small multiples was revised to reflect a blend of the colors used in the first two columns, enhancing participants' understanding of the data representation.

User-oriented Design: In situations where specific users have requirements that can not be addressed by RSVP, a traditional design study may be warranted. Nonetheless, RSVP continues to play a crucial role by serving as a bridge between domain scientists and visualization researchers, effectively acting as a liaison [91]. Domain scientists can utilize RSVP to construct dashboards with their own data, simplifying the explanation of data, tasks, and system limitations to visualization researchers. Consequently, these researchers can better grasp the actual needs of domain scientists and expedite the design of potential solutions.

RSVP was designed with extensibility in mind for such cases. It provides a mechanism whereby visualization designers proficient in creating visualizations with Vega can develop new visualizations. As long as these visualizations adhere to certain requirements regarding data setup, RSVP can seamlessly integrate them into the visualization dashboard (VDB). This integration has the potential to substantially reduce the time required for design studies, which in some instances can extend over several years [77]. Although this mechanism within RSVP is still in its early stages, it will be refined in future iterations of the system.

Comparison to Tableau: The draggable markers in RSVP's data-selection panel (DSP) were inspired by Tableau. However, Tableau does not allow for editing two different visualizations simultaneously, unlike RSVP, which facilitates this through two distinct spatial fields. This is a critical aspect when creating VPSA applications, as input and output dimensions are often encoded in separate views.

Similar to VisRec in RSVP, Tableau's "Show Me" feature also utilizes small multiples of different visualizations, employs colored frames to highlight recommended visualizations, and provides hints to the user on how to encode a specific visualization. However, the small multiples in Tableau are static icons and do not display actual user data. To view them encoded, the user must instantiate a visualization. If a user wishes to compare multiple visualizations, they must instantiate them individually and compare them sequentially. In RSVP, all available visualizations are instantiated simultaneously, allowing for easy comparison. This approach aims to address potential issues with visualization literacy and save time for users, which is particularly important for domain scientists. Furthermore, recommendations in Tableau are generic suggestions based on certain data features, while recommendations in RSVP are task-oriented, tailored to problems commonly encountered in VPSA.

Tableau can handle 1D objects in the form of additionally loaded and linked data sheets. 2D objects can be added as links to images, similar to how it is done in RSVP. However, converting links into images is not straightforward in Tableau and requires considerable expertise to accomplish.

This becomes even more challenging when juxtaposing or superimposing selected images. Furthermore, Tableau does not provide guidance on combining these complex objects with other visualizations in the dashboard to facilitate analysis and achieve specific high-level tasks.

In summary, Tableau offers greater overall power compared to RSVP. However, RSVP is highly efficient and effective for handling VPSA-related problems. This comparison is akin to comparing a domain-specific language (DSL) with a general-purpose programming language: while a general-purpose language can theoretically achieve anything, it may be complex to achieve. In contrast, a DSL simplifies achieving specific tasks by making operations irrelevant to the domain impossible [92].

6.3 Limitations and Future Work

Number of Participants in Usability Study: Our decision for $n = 6$ follows recommendations aimed at optimizing sample size for usability tests to maximize return on investment (ROI) [86], [87]. We set our problem discovery goal at a value of 0.9, which we considered sufficient for a prototype at this stage. Based on a preliminary study involving two users and a previous case study, we estimated a problem discovery rate $p = 0.33$ per user. In order to meet our goal with this rate, six participants were necessary [86]. If we wanted to enhance the robustness of the results, $n = 8$ could accommodate variability in the discovery rate up to 0.08 points. In other words, if our estimated discovery rate was overly optimistic at 0.33, but the actual rate was 0.25, $n = 8$ would achieve our goal of 0.9. Conversely, if our estimation of 0.33 was accurate, $n = 8$ would increase the problem discovery probability to 0.95. These considerations pertain to the evaluation of a single prototype. If the SUS was used to assess user preferences between two competing prototypes, eight to twelve participants would be necessary to ensure robust results [93]. However, since our study currently does not involve a comparison between prototypes, this recommendation does not immediately apply to our situation.

Predictions: The dataflow model presented in Figure 2a is an abbreviated version of the model proposed by Sedlmair et al. The complete version includes an alternative prediction step, which allows analyzing data points that have not been sampled and simulated yet but were approximated with a surrogate model. Using prediction allows the continuous analysis of the parameter space. However, it requires more complicated visualizations and advanced navigation techniques, and keeping selections consistent across discrete and continuous views would be rather complicated, if possible at all. Showing complex objects for non-simulated data points would be another challenge. Training deep learning based surrogate models capable of approximating complex objects for not simulated points can take up to several hours on a supercomputer [94], which is not feasible for most users. Despite these challenges, including predictive analysis capabilities in a generic visualization tool would undoubtedly appeal to many users performing VPSA.

Transferability: We have shown how a meta design study can help extract a visualization design space and a suitable visualization recommendation strategy for VPSA.

However, we believe this approach can be applied to other problem domains For example, many of the papers analyzed in our study deal with complex objects and devote considerable effort to describing techniques for transforming them into lower-dimensional representations for analysis. Selecting appropriate transformations can be challenging, and users may not be aware of all possible transformations for their analyzed objects. To the best of our knowledge, assisting users in selecting suitable transformations for complex objects into alternative representations offering enhanced analytical capabilities is currently a research gap that could potentially be addressed by a meta design study.

Scalablity: As previously stated, VPSA is an approach that facilitates the analysis of multi-dimensional input-output-related problems, where the dimensions carry semantic meaning. For high-dimensional problems encountered in fields like machine learning and deep learning, alternative strategies exist Hohman et al., 2019. There is no definitive threshold distinguishing between multi-dimensional and high-dimensional problems [9], but the analysis of the applications in the meta design study revealed that most of them featured examples with less than ten dimensions, and none of them exceeded twenty.

RSVP provides visualization options and recommendations for up to fifteen input and output dimensions, respectively, with support for up to one thousand runs (Con-2). If these limits are significantly exceeded, alternative visualization options and a potentially faster rendering solutions (such as a WebGL/WebGPU implementation of the renderer in Vega) may be necessary.

Extensibility: As previously explained, the visualization dashboard is already extensible with new Vega visualizations (though currently undocumented), but the visualization options in the overview area are not currently extensible. Additionally, the VisRec rules are presently derived manually, which complicates the introduction of new rules or updates to existing ones when new papers or applications are added to the knowledge base. Converting our hard-coded rules into constraints for answer-set-programming or utilizing a framework like Knowledge Rocks for handling visualization knowledge might mitigate this problem, but further research is necessary in this regard. Moreover, automating the extraction of new VisRec rules from design study papers is a task for future research.

Sampling Support: Users often can sample their model using random or Cartesian grid sampling but are incapable of using more sophisticated strategies like Latin Hypercube sampling [95] or uniform sampling on a space-filling curve [96]. However, advanced visualization techniques like the previously discussed ones also necessitate improved sampling strategies. Furthermore, Latin Hypercube Sampling would probably improve uncertainty and sensitivity analysis [97]. We envision a tool like Hanpuku [98], which would support domain scientists in sampling their models with more sophisticated strategies and create outputs that could be directly used by tools like RSVP.

Automated Dashboard Design: RSVP provides multi-view recommendations and hints to users on how to interact with them, but it does not support arranging those views in the dashboard. The size of individual views related to the amount of data they encode could be determined using Pixnostics [99], Scagnostics [100], and Pargnostics [101]. Layout proposals could be based on simple heuristics [102], or they could be learned from online dashboards [103]. A domain-oriented recommendation approach could also be considered. Domain scientists tag and share dashboards they deem helpful, and the system could recommend those to users facing a similar problem. The similarity could be established by analyzing those tags and data [104] or via an ontology-based approach [58], [105].

7 CONCLUSION

We presented RSVP, a system based on an extensive meta design study on the subject of VPSA, designed to enable domain scientists wihtout programming skills to create custom visualization dashboards quickly. It features a task-oriented visualization recommendation strategy and a drag-and-drop interaface for creating said dashbaords. RSVP not only guides users through their data analysis needs but also promotes learning and provides explanations for its recommendations. A usability study and case studies confirm RSVP's user-friendliness and its ability to provide valuable insights, suggesting its applicability across various fields. Future updates will focus on enhancing RSVP's capabilities regarding the expansion of visualization options, recommendations, and support for more complex data types.

REFERENCES

[1] K. Potter, A. Wilson, P. Bremer, D. Williams, C. Doutriaux, V. Pascucci, and C. R. Johnson, "Ensemble-Vis: A framework for the statistical visualization of ensemble data," in *IEEE International Conference on Data Mining Workshops*, 2009, pp. 233–240.

[2] J. Wang, X. Liu, H. Shen, and G. Lin, "Multi-resolution climate ensemble parameter analysis with nested parallel coordinates plots," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 81–90, 2017.

[3] K. Matkovic, D. Gracanin, B. Klarin, and H. Hauser, "Interactive visual analysis of complex scientific data as families of data surfaces," *IEEE Transactions on Visualization and Computer Graphics*, 2009.

[4] L. Cibulski, H. Mitterhofer, T. May, and J. Kohlhammer, "Paved: Pareto front visualization for engineering design," *Computer Graphics Forum*, vol. 39, no. 3, pp. 405–416, 2020.

[5] S. Bruckner and T. Möller, "Result-driven exploration of simulation parameter spaces for visual effects design," *IEEE Transactions on Visualization and Computer Graphics*, 2010.

[6] J. Matejka, M. Glueck, E. Bradner, A. Hashemi, T. Grossman, and G. Fitzmaurice, "Dream Lens: Exploration and visualization of large-scale generative design datasets," in *CHI Conference on Human Factors in Computing Systems*, ser. CHI '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1–12.

[7] T. Mühlbacher, L. Linhardt, T. Möller, and H. Piringer, "TreePOD: Sensitivity-aware selection of pareto-optimal decision trees," *IEEE Transactions on Visualization and Computer Graphics*, 2018.

[8] S. Hamid, A. Derstroff, S. Klemm, Q. Q. Ngo, X. Jiang, and L. Linsen, "Visual ensemble analysis to study the influence of hyper-parameters on training deep neural networks," in *Machine Learning Methods in Visualisation for Big Data*. The Eurographics Association, 2019.

[9] M. Sedlmair, C. Heinzl, S. Bruckner, H. Piringer, and T. Möller, "Visual parameter space analysis: A conceptual framework," *IEEE Transactions on Visualization and Computer Graphics*, 2014.

[10] M. Sedlmair, M. Meyer, and T. Munzner, "Design study methodology: Reflections from the trenches and the stacks," *IEEE Transactions on Visualization and Computer Graphics*, 2012.

[11] E. Horvitz, "Principles of mixed-initiative user interfaces," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 1999.

[12] P. Federico, M. Wagner, A. Rind, A. Amor-Amorós, S. Miksch, and W. Aigner, "The role of explicit knowledge: A conceptual model of knowledge-assisted visual analytics," in *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2017.

[13] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[14] A. J. Pretorius, M. Bray, A. E. Carpenter, and R. A. Ruddle, "Visualization of parameter space for image analysis," *IEEE Transactions on Visualization and Computer Graphics*, 2011.

[15] T. Torsney-Weir, A. Saad, T. Möller, H. Hege, B. Weber, J. Verbavatz, and S. Bergner, "Tuner: Principled parameter finding for image segmentation algorithms using visual response surface exploration," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 1892–1901, 2011.

[16] B. Fröhler, T. Möller, and C. Heinzl, "GEMSe: Visualization-guided exploration of multi-channel segmentation algorithms," *Computer Graphics Forum*, vol. 35, no. 3, pp. 191–200, 2016.

[17] M. Oppermann and T. Munzner, "Data-first visualization design studies," in *IEEE Workshop on Evaluation and Beyond - Methodological Approaches to Visualization (BELIV)*, 2020.

[18] M. W. Lipsey and D. B. Wilson, *Practical meta-analysis*. SAGE publications, Inc, 2001.

[19] T. Munzner, "A nested model for visualization design and validation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 15, no. 6, pp. 921–928, 2009.

[20] T. Munzner, *Visualization analysis and design*. CRC press, 2014.

[21] M. Sedlmair, P. Isenberg, D. Baur, and A. Butz, "Information visualization evaluation in large companies: Challenges, experiences and recommendations," *Information Visualization*, 2011.

[22] D. M. Russell, "Simple is good: Observations of visualization use amongst the big data digerati," in *International Working Conference on Advanced Visual Interfaces*, ser. AVI '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 7–12.

[23] N. Boukhelifa, A. Bezerianos, I. C. Trelea, N. M. Perrot, and E. Lutton, "An exploratory study on visual exploration of model simulations by multiple types of experts," in *CHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 2019.

[24] D. Ma'ayan, W. Ni, K. Ye, C. Kulkarni, and J. Sunshine, "How domain experts create conceptual diagrams and implications for tool design," in *CHI Conference on Human Factors in Computing Systems*, ser. CHI '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 1–14.

[25] A. Slingsby and J. Dykes, "Experiences in involving analysts in visualisation design," in *Proceedings of the 2012 BELIV Workshop: Beyond Time and Errors - Novel Evaluation Methods for Visualization*, New York, NY, USA, 2012.

[26] Y. Ye, F. Sauer, K.-L. Ma, K. Aditya, and J. Chen, "A user-centered design study in scientific visualization targeting domain experts," *IEEE Transactions on Visualization and Computer Graphics*, 2020.

[27] E. Mayr, P. Federico, S. Miksch, G. Schreder, M. Smuc, and F. Windhager, "Visualization of cultural heritage data for casual users," in *IEEE VIS Workshop on Visualization for the Digital Humanities*, 2016.

[28] A. Gegenfurtner and M. Seppänen, "Transfer of expertise: An eye tracking and think aloud study using dynamic medical visualizations," *Computers & Education*, 2013.

[29] P. Ruchikachorn and K. Mueller, "Learning visualizations by analogy: Promoting visual literacy through visualization morphing," *IEEE Transactions on Visualization and Computer Graphics*, vol. 21, no. 9, pp. 1028–1044, 2015.

[30] Y. L. Wong, K. Madhavan, and N. Elmqvist, "Towards characterizing domain experts as a user group," in *IEEE Evaluation and Beyond - Methodological Approaches for Visualization (BELIV)*, 2018.

[31] L. Grammel, "User interfaces supporting information visualization novices in visualization construction," Ph.D. dissertation, 2012.

[32] J. van Wijk, "The value of visualization," in *IEEE Visualization*, 2005.

[33] A. Dasgupta, J. Lee, R. Wilson, R. A. Lafrance, N. Cramer, K. Cook, and S. Payne, "Familiarity vs trust: A comparative study of domain scientists' trust in visual analytics and conventional analysis methods," *IEEE Transactions on Visualization and Computer Graphics*, 2017.

[34] S. Zhu, G. Sun, Q. Jiang, M. Zha, and R. Liang, "A survey on automatic infographics and visualization recommendations," *Visual Informatics*, vol. 4, no. 3, pp. 24–40, 2020.

[35] J. Mackinlay, "Automating the design of graphical presentations of relational information," *ACM Transactions on Graphics*, 1986.

[36] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer, "Towards a general-purpose query language for visualization recommendation," in *Proceedings of the Workshop on Human-In-the-Loop Data Analytics*. New York, NY, USA: Association for Computing Machinery, 2016.

[37] K. Wongsuphasawat, Z. Qu, D. Moritz, R. Chang, F. Ouk, A. Anand, J. Mackinlay, B. Howe, and J. Heer, "Voyager 2: Augmenting visual analysis with partial view specifications," in *CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2017.

[38] D. Moritz, C. Wang, G. L. Nelson, H. Lin, A. M. Smith, B. Howe, and J. Heer, "Formalizing visualization design knowledge as constraints: Actionable and extensible models in draco," *IEEE Transactions on Visualization and Computer Graphics*, 2019.

[39] H. Lin, D. Moritz, and J. Heer, "Dziban: Balancing agency & automation in visualization design via anchored recommendations," in *CHI Conference on Human Factors in Computing Systems*, 2020.

[40] S. Wehrend and C. Lewis, "A problem-oriented classification of visualization techniques," in *IEEE Conference on Visualization: Visualization '90*, 1990, pp. 139–143.

[41] "Task-analytic approach to the automated design of graphic presentations," *ACM Trans. Graph.*, 1991.

[42] S. F. Roth, J. Kolojejchick, J. Mattis, and J. Goldstein, "Interactive graphic design using automatic presentation knowledge," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1994, pp. 112–117.

[43] M. X. Zhou and S. K. Feiner, "Visual task characterization for automated visual discourse synthesis," in *CHI Conference on Human Factors in Computing Systems*, ser. CHI '98. USA: ACM Press/Addison-Wesley Publishing Co., 1998, p. 392–399.

[44] R. Amar and J. Stasko, "A knowledge task-based framework for design and evaluation of information visualizations," in *IEEE Symposium on Information Visualization*, 2004, pp. 143–150.

[45] C. Demiralp, P. J. Haas, S. Parthasarathy, and T. Pedapati, "Foresight: Recommending visual insights," *Proc. VLDB Endow.*, vol. 10, no. 12, p. 1937–1940, Aug. 2017.

[46] D. J.-L. Lee, V. Setlur, M. Tory, K. G. Karahalios, and A. Parameswaran, "Deconstructing categorization in visualization recommendation: A taxonomy and comparative study," *IEEE Transactions on Visualization and Computer Graphics*, 2021.

[47] L. Shen, E. Shen, Z. Tai, Y. Song, and J. Wang, "Taskvis: Task-oriented visualization recommendation," 2021.

[48] A. Key, B. Howe, D. Perry, and C. Aragon, "VizDeck: Self-organizing dashboards for visual analytics," in *ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '12, New York, NY, USA, 2012, p. 681–684.

[49] Y. Luo, X. Qin, N. Tang, and G. Li, "DeepEye: Towards automatic data visualization," in *IEEE International Conference on Data Engineering (ICDE)*, 2018, pp. 101–112.

[50] A. Wu, Y. Wang, M. Zhou, X. He, H. Zhang, H. Qu, and D. Zhang, "Multivision: Designing analytical dashboards with deep learning based recommendation," *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2021.

[51] N. Tintarev and J. Masthoff, "A survey of explanations in recommender systems," in *2007 IEEE 23rd International Conference on Data Engineering Workshop*, 2007, pp. 801–810.

[52] A. McNutt, A. Crisan, and M. Correll, "Divining insights: Visual analytics through cartomancy," in *CHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 2020.

[53] H. Schulz, T. Nocke, M. Heitzler, and H. Schumann, "A design space of visualization tasks," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 12, pp. 2366–2375, 2013.

[54] H. Li, Y. Wang, S. Zhang, Y. Song, and H. Qu, "Kg4vis: A knowledge graph-based approach for visualization recommendation," *IEEE Transactions on Visualization and Computer Graphics*, 2021.

[55] K. Hu, M. A. Bakker, S. Li, T. Kraska, and C. Hidalgo, *VizML: A Machine Learning Approach to Visualization Recommendation*, New York, NY, USA, 2019.

[56] O. Gilson, N. Silva, P. Grant, and M. Chen, "From web data to visualization via ontology mapping," *Computer Graphics Forum*, 2008.

[57] J. Polowinski and M. Voigt, "Viso: A shared, formal knowledge base as a foundation for semi-automatic infovis systems," in *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, New York, NY, USA, 2013.

[58] A.-P. Lohfink, S. D. D. Anton, H. Leitte, and C. Garth, "Knowledge rocks: Adding knowledge assistance to visualization systems," *IEEE Transactions on Visualization and Computer Graphics*, 2021.

[59] K. Wongsuphasawat, D. Moritz, A. Anand, J. 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, 2016.

[60] E. Chi, P. Barry, J. Riedl, and J. Konstan, "A spreadsheet approach to information visualization," in *Proceedings of VIZ '97: Visualization Conference, Information Visualization Symposium and Parallel Rendering Symposium*, 1997, pp. 17–24.

[61] T. J. Jankun-Kelly and Kwan-Liu Ma, "Visualization exploration and encapsulation via a spreadsheet-like interface," *IEEE Transactions on Visualization and Computer Graphics*, 2001.

[62] J. Roberts, "On encouraging multiple views for visualization," in *Proceedings. IEEE Conference on Information Visualization*, 1998.

[63] S. van den Elzen and J. J. van Wijk, "Small multiples, large singles: A new approach for visual data exploration," *Computer Graphics Forum*, 2013.

[64] C. Weaver, "Building highly-coordinated visualizations in Improvise," in *IEEE Symposium on Information Visualization*, 2004.

[65] K. Matkovic, W. Freiler, D. Gracanin, and H. Hauser, "ComVis: A coordinated multiple views system for prototyping new visualization technology," in *International Conference Information Visualisation*, 2008.

[66] H. Piringer, C. Tominski, P. Muigg, and W. Berger, "A multithreading architecture to support interactive visual exploration," *IEEE Transactions on Visualization and Computer Graphics*, 2009.

[67] M. A. Yalçın, N. Elmqvist, and B. B. Bederson, "Keshif: Rapid and expressive tabular data exploration for novices," *IEEE Transactions on Visualization and Computer Graphics*, 2018.

[68] "Tableau," https://www.tableau.com, accessed: 2024-06-28.

[69] C. Stolte, D. Tang, and P. Hanrahan, "Polaris: a system for query, analysis, and visualization of multidimensional relational databases," *IEEE Transactions on Visualization and Computer Graphics*, vol. 8, no. 1, pp. 52–65, 2002.

[70] A. Bryman, *Social research methods*. Oxford university press, 2016.

[71] D. Furniss, A. Blandford, and P. Curzon, "Confessions from a grounded theory phd: Experiences and lessons learnt," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 113–122.

[72] A. MacLean, R. M. Young, and T. P. Moran, "Design rationale: The argument behind the artifact," in *SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '89. New York, NY, USA: Association for Computing Machinery, 1989, p. 247–252.

[73] A. Unger, S. Schulte, V. Klemann, and D. Dransch, "A visual analysis concept for the validation of geoscientific simulation models," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2216–2225, 2012.

[74] H. Piringer, W. Berger, and J. Krasser, "HyperMoVal: Interactive visual validation of regression models for real-time simulation," *Computer Graphics Forum*, vol. 29, no. 3, pp. 983–992, 2010.

[75] J. Weissenböck, A. Amirkhanov, E. Gröller, J. Kastner, and C. Heinzl, "PorosityAnalyzer: Visual analysis and evaluation of segmentation pipelines to determine the porosity in fiber-reinforced polymers," in *IEEE Conference on Visual Analytics Science and Technology (VAST)*, 2016, pp. 101–110.

[76] W. Berger, H. Piringer, P. Filzmoser, and E. Gröller, "Uncertainty-aware exploration of continuous parameter spaces using multivariate prediction," *Computer Graphics Forum*, 2011.

[77] M. Booshehrian, T. Möller, R. M. Peterman, and T. Munzner, "Vismon: Facilitating analysis of trade-offs, uncertainty, and sensitivity in fisheries management decision making," *Computer Graphics Forum*, vol. 31, no. 3pt3, pp. 1235–1244, 2012.

[78] M. Luboschik, S. Rybacki, F. Haack, and H.-J. Schulz, "Supporting the integrated visual analysis of input parameters and simulation trajectories," *Computers & Graphics*, vol. 39, pp. 37 – 47, 2014.

[79] A. Satyanarayan and J. Heer, "Lyra: An interactive visualization design environment," *Computer Graphics Forum*, 2014.

[80] B. Shneiderman, "The eyes have it: a task by data type taxonomy for information visualizations," in *IEEE Symposium on Visual Languages*, 1996.

[81] J. C. Roberts, "Encouraging coupled views for visualization exploration," in *Visual Data Exploration and Analysis VI*, R. F. Erbacher, P. C. Chen, and C. M. Wittenbrink, Eds. International Society for Optics and Photonics, 1999.

[82] S. Lee, S. Kim, Y. Hung, H. Lam, Y. Kang, and J. S. Yi, "How do people make sense of unfamiliar visualizations?: A grounded model of novice's information visualization sensemaking," *IEEE Transactions on Visualization and Computer Graphics*, 2016.

[83] D. Ceneda, T. Gschwandtner, T. May, S. Miksch, H. Schulz, M. Streit, and C. Tominski, "Characterizing guidance in visual analytics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 1, pp. 111–120, 2017.

[84] M. Khayat, M. Karimzadeh, D. S. Ebert, and A. Ghafoor, "The validity, generalizability and feasibility of summative evaluation methods in visual analytics," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 353–363, 2020.

[85] T. Isenberg, P. Isenberg, J. Chen, M. Sedlmair, and T. Möller, "A systematic review on the practice of evaluating visualization," *IEEE Transactions on Visualization and Computer Graphics*, 2013.

[86] C. W. Turner, J. R. Lewis, and J. Nielsen, "Determining usability test sample size," *International encyclopedia of ergonomics and human factors*, vol. 3, no. 2, pp. 3084–3088, 2006.

[87] J. R. Lewis, "Sample sizes for usability studies: Additional considerations," *Human Factors*, 1994.

[88] A. Dasgupta, S. Burrows, K. Han, and P. J. Rasch, "Empirical analysis of the subjective impressions and objective measures of domain scientists' visual analytic judgments," in *Proceedings CHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 2017.

[89] J. Brooke, "Sus: a retrospective," *Journal of usability studies*, 2013.

[90] C. Kittel and P. McEuen, *Introduction to solid state physics*. John Wiley & Sons, 2018.

[91] S. Simon, S. Mittelstädt, D. A. Keim, and M. Sedlmair, "Bridging the gap of domain and visualization experts with a liaison," in *Eurographics Conference on Visualization (EuroVis) - Short Papers*, 2015.

[92] A. M. McNutt, "No grammar to rule them all: A survey of json-style dsls for visualization," *IEEE Transactions on Visualization and Computer Graphics*, 2023.

[93] T. S. Tullis and J. N. Stetson, "A comparison of questionnaires for assessing website usability," in *Usability professional association conference*, 2004.

[94] W. He, J. Wang, H. Guo, K.-C. Wang, H.-W. Shen, M. Raj, Y. S. G. Nashed, and T. Peterka, "Insitunet: Deep image synthesis for parameter space exploration of ensemble simulations," *IEEE Transactions on Visualization and Computer Graphics*, 2020.

[95] M. D. Mckay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, 2000.

[96] Z. He and A. B. Owen, "Extensible grids: uniform sampling on a space filling curve," *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 2016.

[97] J. Helton and F. Davis, "Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems," *Reliability Engineering & System Safety*, 2003.

[98] A. Bigelow, S. Drucker, D. Fisher, and M. Meyer, "Iterating between tools to create and edit visualizations," *IEEE Transactions on Visualization and Computer Graphics*, 2017.

[99] J. Schneidewind, M. Sips, and D. A. Keim, "Pixnostics: Towards measuring the value of visualization," in *2006 IEEE Symposium On Visual Analytics Science And Technology*, 2006, pp. 199–206.

[100] L. Wilkinson, A. Anand, and R. Grossman, "Graph-theoretic scagnostics," in *Information Visualization, IEEE Symposium on*. IEEE Computer Society, 2005, pp. 21–21.

[101] A. Dasgupta and R. Kosara, "Pargnostics: Screen-space metrics for parallel coordinates," *IEEE Transactions on Visualization and Computer Graphics*, 2010.

[102] B. Bach, Z. Wang, M. Farinella, D. Murray-Rust, and N. Henry Riche, "Design patterns for data comics," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 2018.

[103] Y. Lin, H. Li, A. Wu, Y. Wang, and H. Qu, "Dminer: Dashboard design mining and recommendation," 2022.

[104] M. Oppermann, R. Kincaid, and T. Munzner, "Vizcommender: Computing text-based similarity in visualization repositories for content-based recommendations," *IEEE Transactions on Visualization and Computer Graphics*, 2021.

[105] T. Sobral, T. Galvão, and J. Borges, "An ontology-based approach to knowledge-assisted integration and visualization of urban mobility data," *Expert Systems with Applications*, 2020.

Michael Wimmer is currently a full professor at the Institute of Visual Computing and Human-Centered Technology at TU Wien, where he heads the Rendering and Modeling Group. He is also the director of the Center for Geometry and Computational Design (GCD). His academic career started with his M.Sc. in 1997 at TU Wien, where he also obtained his Ph.D. in 2001. His research interests are real-time rendering, computer games, real-time visualization of urban environments, point-based rendering, reconstruction of urban models, procedural modeling, shape modeling, and computational design.

Manfred Klaffenboeck is a doctoral candidate at TU Wien's Institute for Visual Computing and Human-Centered Technology in Vienna, Austria. He earned a Magister degree in Movie and Media Science in 2013, followed by a BSc in Computer Science in 2016 and an MSc in Media Informatics in 2018, all from the University of Vienna. His research integrates computer graphics, computer vision, data visualization, and media analysis, with a current focus on semi-automated multiverse analysis leveraging (visual) parameter space exploration, visual analytics, and knowledge assistance.

Torsten Möller (Senior Member, IEEE) received the Vordiplom (BSc) degree in mathematical computer science from the Humboldt University of Berlin, Germany, and the PhD in computer and information science from Ohio State University in 1999. He has been a professor of computer science at the University of Vienna, Austria, since 2013. Between 1999 and 2012, he served as a computing science faculty member at Simon Fraser University, Canada. He is a senior member of ACM, IEEE, and a member of Eurographics. His research interests include algorithms and tools for analyzing and displaying data with principles rooted in computer graphics, human-computer interaction, signal processing, data science, and visualization.

Michael Gleicher is a Professor in the Department of Computer Sciences at the University of Wisconsin, Madison. Prof. Gleicher is founder of the Department's Visual Computing Group and co-directs both the Visual Computing Laboratory and the Collaborative Robotics Laboratory at UW-Madison. His research interests span the range of visual computing, including data visualization, robotics, and virtual/extended reality. His recent work includes exploring perceptual issues in visualization, the use of visual simulation for robotics, and geometric approaches to enhance robot perception and interaction. He earned his Ph. D. in Computer Science (1994) from Carnegie Mellon University, and earned a B.S.E. in Electrical Engineering from Duke University (1988). In 2023-2024, Prof. Gleicher holds a concurrent appointment as a Design Scholar at Amazon Robotics. This work is not associated with Amazon.

Johannes Sorger received his computer science degree in visual computing from the TU Wien at the Institute of Computer graphics and Algorithms, where he also finished his PhD in collaboration with the VRVis research center. Johannes' main research interests are centred around the application of visualization as an enabling technology. For his work on the visualization of neuronal networks Johannes received the 2014 OCG Incentive Award, as well as the Best Paper Award at the 2013 IEEE Symposium on Biological Data Visualization. During his postdoc, he was heading the visualization team at CSH Vienna where he researched novel approaches in Immersive network analytics.