Scalability in Visualization

Gaëlle Richer, Alexis Pister, Moataz Abdelaal, Jean-Daniel Fekete, Michael Sedlmair, and Daniel Weiskopf

Abstract—We introduce a conceptual model for scalability designed for visualization research. With this model, we systematically analyze over 120 visualization publications from 1990 to 2020 to characterize the different notions of scalability in these works. While many papers have addressed scalability issues, our survey identifies a lack of consistency in the use of the term in the visualization research community. We address this issue by introducing a consistent terminology meant to help visualization researchers better characterize the scalability aspects in their research. It also helps in providing multiple methods for supporting the claim that a work is "scalable." Our model is centered around an effort function with inputs and outputs. The inputs are the problem size and resources, whereas the outputs are the actual efforts, for instance, in terms of computational run time or visual clutter. We select representative examples to illustrate different approaches and facets of what scalability can mean in visualization literature. Finally, targeting the diverse crowd of visualization researchers without a scalability tradition, we provide a set of recommendations for how scalability can be presented in a clear and consistent way to improve fair comparison between visualization techniques and systems and foster reproducibility.

Index Terms—Scalability, visualization, structured literature analysis, conceptual framework

✦

1 INTRODUCTION

We address the issue of characterizing *scalability* in visualization research. Scalability is a frequent topic, with many papers claiming to improve scalability or achieve scalable—or sometimes, more scalable—techniques. The visualization research community has a long tradition of acknowledging the need for scalable solutions, as for example, included in summaries of grand research challenges for various communities of visualization [1], [2] or roadmaps for future research [3], [4].

Despite the high relevance of scalability—or maybe, because of this—we noticed a large range of connotations or uses of this concept in visualization papers. This situation reflects the large diversity of research topics and methods in visualization, and it may also be the multidisciplinary nature of visualization, which includes research from computer science and algorithms, human-computer interaction, psychology, etc. Some of these communities have established models and methods for assessing scalability, but not all of them. However, even when such approaches might be established in another community, they might not necessarily be common knowledge in the visualization research community. Furthermore, it is not always clear whether a wholesale adoption of these methods is possible or if they need to be adapted and fine-tuned to the specificities of visualization research.

In short, there is a wide range of interpretations of the concept of scalability in visualization, sometimes with only implicit documentation and communication of the concrete interpretation used in a paper. This can lead to misunderstandings and impair the reproducibility of research results.

- G. Richer, A. Pister, and J.-D. Fekete are with Université Paris-Saclay, CNRS, Inria, LISN, France. A. Pister is also with I3, CNRS, Telecom Paris, Institut Polytechnique de Paris, France.
  E-mail: {first-name.last-name}@inria.fr
- M. Adbelaal, M. Sedlmair, and D. Weiskopf are with University of Stuttgart, Germany.
  E-mail: {first-name.last-name}@visus.uni-stuttgart.de

The recent restructuring of the IEEE VIS conferences into a single conference with multiple areas attests that visualization research is becoming more diverse and trying to be more integrated. While some articles will remain targeted to a distinct audience well aware of its own meaning of scalability, a growing number of articles will cross boundaries to address multiple meanings of scalability, leading to more diverse reviewers and readers, with different backgrounds. We aim at helping authors, reviewers, and readers navigate the different aspects of scalability.

To this end, we contribute a *conceptual model for scalability* that is designed to be versatile and flexible enough to capture existing uses of the concept 'scalability' in visualization research, align terminology, improve conceptual and methodological consistency across domains, and allow for other uses in the future. In particular, we envision the model to help communicate about scalability across the diverse subcommunities of visualization. Our model is built on an *effort* function that takes inputs in the form of problem size, assumptions, and descriptions of resources, and maps these to a description of effort as the output associated with the visualization. Key to the flexibility of the model is the large freedom in modeling the inputs and outputs: they can cover technical aspects such as data set size, available compute nodes, or compute times, all the way to human-oriented aspects like readability or user task performance. Therefore, we are able to show that this model can be instantiated to cover the typical scenarios of scalability in visualization, and also the different interpretations of the terms "scalability," "scalable," and "more scalable."

We argue that seeking the common traits between the multiple existing definitions and presenting them with a unified model creates a helpful framework for comprehension. We recognize our model would not help authors and reviewers within their subfield, *e.g.*, visualization in high-performance computing (HPC) or graph drawing, since they have a clear understanding of their own meaning of scalability. However, it becomes useful for articles mixing two aspects of scalability, *e.g.*, how HPC can provide more readable features, with a need to be understood by the two subcommunities. This kind of scenario is becoming more frequent in visualization and this is why we need a unifying model.

Using the conceptual model as a framework, we analyzed the current state of visualization research and contribute a *structured and systematic literature analysis* of the full papers published in IEEE Visualization, SciVis, InfoVis, and VAST from 1990 to 2020. The literature search led to 127 articles for which we derived a coding scheme and analyzed them, respectively. Four of the authors participated in multiple rounds of reviews of these relevant papers followed by discussions to establish the conceptual model, scenarios, and coding scheme. The two other authors coded the complete set of papers after being introduced to the coding scheme. Our goal was to learn about the current usage of the notion of scalability in visualization research, as well as to assess how well our conceptual model allows characterizing previous research on scalability. We make the coding book and results publicly available at the following repository: `https://osf.io/xrvu7/`.

Based on our conceptual model, general observations, and the literature review, we arrive at *recommendations* to improve the design and presentation of scalability-related research when targeting an outside or mixed audience. We believe that this would also help compare visualization techniques and systems, and foster reproducibility.

2 RELATED WORK

The visualization research community has become more diverse over the years, starting with statistics, algorithms, computer graphics, and computational science in the early 1990s, and joined by human-computer interaction (HCI), psychology, vision science, design, cartography, and many more. The concept of scalability varies from one community to the next, with different levels of maturity. In this section, we review related work discussing and defining scalability in different areas of computer science and in the visualization community.

2.1 Definitions of Scalability

Weinstock and Goodenough [5] define the scalability problem as "the inability of a system to accommodate an increased workload." Bondi [6] mentions several definitions of scalability in computer science:

- "*Scalability* is the property of a system to handle a growing amount of work by adding resources to the system." Adding resources may have the form of adding more nodes to a system made of multiple small interconnected servers (scaling *out* or *horizontally*) or adding more resources to a single node (scaling *up* or *vertically*) [7].
- *Load scalability* is the "ability to function gracefully, *i.e.*, without undue delay and without unproductive resource consumption or resource contention at light, moderate, or heavy loads while making good use of available resources."
- *Space scalability* is that "memory requirements do not grow to intolerable levels as the number of items it supports increase."
- *Space-time scalability*: "continues to function gracefully as the number of objects [. . .] increases by orders of magnitude."
- *Structural scalability* means that "implementation or standards do not impede the growth of the number of objects it encompasses, or at least will not do so within a chosen time frame."

Parallel systems and HPC distinguish mainly two types of scalability:

- *Strong scaling*: "how the solution time varies with the number of processors for a fixed total problem size."

- *Weak scaling*: "how the solution time varies with the number of processors for a fixed problem size per processor."

Hill [8] tries to define scalability for multiprocessor systems and admits: "but I fail to find a useful, rigorous definition of it." Duboc *et al.* [9] define it as: "a quality of software systems characterized by the causal impact that scaling aspects of the system environment and design have on certain measured system qualities as these aspects are varied over expected operational ranges. If the system can accommodate this variation in a way that is acceptable to the stakeholder, then it is a scalable system."

All the definitions are specified as properties of systems at an abstract level, focusing on "amount of work," "delay," "resources," "productive resource consumption," "[work]loads," "memory," "function gracefully," "time frame," "adding nodes," and "shared memory." They rely on implicit domain knowledge to be clearly understood and are not suitable to the wide audience of visualization practitioners.

2.2 Scalability in Visualization and Visual Analytics

Visualization and visual analytics are concerned with general computer science scalability when it comes to systems or algorithms. In addition, they are also concerned with more specific issues. Robertson *et al.* [10] mention information scalability, visual scalability, display scalability, and human scalability, in addition to computational scalability. They also add other scalability issues: software scalability, temporal scalability, cross-scale issues, privacy and security issues (related to scale), and language issues. Yost and North [11] also mention graphical scalability ("limits imposed by the number of pixels") and perceptual scalability ("When the screen is not the limiting factor, just how much data can a person effectively perceive?"). Eick and Karr [12] want to quantify visual scalability by modeling the dependence between responses, factors, and data. They admit that it cannot be done because few responses can be quantified or measured. Instead, they break down the problem into subparts affecting the overall scalability, adding "visual metaphors," "interactivity," and "aggregation" to the list of factors affecting scalability.

Scalability is also related to evaluation since it is based on measuring efficiency. Lam *et al.* [13] describe seven scenarios for evaluation in visualization, some of them leading to quantitative results and others to qualitative ones. Scalability is part of the "Evaluating User Performance" and "Evaluating Visualization Algorithms" scenarios. One area in which scalability evaluation is well-established is the HPC/visualization community, where the main focus is on algorithmic scalability with well-defined metrics and definitions (*e.g.*, strong scalability). However, the rest of the visualization community may not be familiar with these definitions, and it remains unclear if they could be applied in a broader context than those with HPC resources.

2.3 Scalability in HCI, Psychology, and Vision Science

Scalability related to humans is different from scalability in computer science. In their seminal book, Card *et al.* [14] describe the human as a processor with numerous capabilities, some of them ruled by laws or models expressible mathematically. Visualization is concerned with several of these capabilities, in particular regarding perceptual scalability, cognitive scalability, and movement. The psychology laws and models often refer to information theory, considering perception and action as communication through capacity-limited channels.

Scalability has been studied for some aspects of visual perception, such as ensemble coding [15], preattentive processing [16], and the limit of the number of colors perceived efficiently [17], Fitts' law [18] for pointing, the scalability of item selection and navigation [19], Hick's Law [20] for reading items, and the scalability of menus [21]. Budiu [22] discusses several issues related to scaling user interfaces: working memory limits, screen size that limits the capacity of the communication channel, and attention limits. Brown *et al*. [23] list scalability challenges in HCI relative to the number of users, the different contexts of use, and the multiplicity of systems and technologies. Therefore, while most of the human capabilities exhibit hard limits, interaction and visualization techniques allow performing tasks with various interpretations of scalability.

2.4 Summary

Scalability is addressed in many ways by the different disciplines and communities related to visualization. Still, they share many concepts but instantiate these concepts with wide variations.

Several articles relate scalability to "factors and certain dependent variables" [9], [12], [24], also called independent variables and measures. Duboc *et al*. [9] also mention *nuisance variables*: "Variables whose effects cannot be completely controlled for or variables that are simply not considered in the experiment design." They also consider the scalability problem as a multi-criteria optimization problem with multiple measures to optimize, combined into a utility function. Although we acknowledge that their model is useful, we believe it is too complicated with respect to the interpretations of scalability as seen in the visualization research community where the measures are not usually combined.

Human capabilities do not scale as nicely as machine ones. Visual perception is limited in scalability by physiological factors, such as the number of cones and rods at the lower level. Some pattern processing allows humans to perform important tasks efficiently (sometimes called "preattentively"), but these perceptual tasks only work under stringent limits. The eye can track the movement of a few moving objects on a screen, but this tracking fails when too many objects cross (visual crowding). Therefore, scalability-related human performance can hardly be assessed on theoretical grounds only, it should usually be checked with experiments. For these reasons, while scalability in the HPC and distributed computing communities has established definitions and evaluation methodologies, these do not directly translate to all parts of a visualization system with a human in the loop.

In this article, we do not define scalability but provide a model to express particular instances of scalability according to the "utility function" of Duboc *et al*. [9].

3 SCALABILITY MODEL

Our first contribution is a conceptual model that describes the scalability of a visualization system, component, or technique. The model is designed to (1) express different scalability concerns that are relevant to visualization applications (*e.g*., visual, perceptual, computational), (2) be applied to different parts of the visualization pipeline, and (3) allow reasoning about different meanings of *scalable* and *scalability*.

3.1 Model Components

The scalability model represents the scalability of a *visualization process* that tackles a specific problem, by a function with four components: *problem size*, *resources*, *assumptions*, and *effort*, which are described in more detail in the coming subsections. The function maps the problem size, expected to vary or grow across applications, to the effort associated with the process's solution to the problem, provided an amount of resources and some assumptions, specific to the particular problem addressed. The relationship between these four components is formalized by the function f:

$$f : (S; R, A) \longmapsto E$$

with S being the set of variables describing the problem size, R describing the available resources, A the assumptions, and E the effort associated with the result. The components of the conceptual model are summarized in Figure 1. The notation separates S from R and A to express the difference in role of the actual input S from the context parameters R and A.

3.1.1 Problem Sizes S

The *problem size* variables are properties that characterize the complexity of the problem targeted or solved by the process. Most commonly, these variables are descriptions of the size of the input data: either in number of elements or attributes for discrete data, or in sample size for continuous data. However, they could also correspond to data characteristics that go beyond data size, such as data distribution, or refer to input other than data, such as the number of simultaneous users or the visual output size (*e.g*., image resolution).

3.1.2 Resources R

The *resource* variables are properties related to the material components of the system or application environment. They are factors influencing effort while being independent of the input data. They typically include computational resources (*e.g*., number of cores, memory), or other resources that the designer can leverage to improve performance. Resources are characterized by the fact that they are often limited in practice, and therefore the optimization of their usage is one lever to improve performance.

In some communities like HPC, being scalable encompasses the intent to optimize resource usage as well as being designed to gracefully adapt and make use of any additional resource available at their maximum capacity. Examples include networks of computers (*e.g*., [25]) or grids of projectors (*e.g*., [26]). In other communities like HCI, having additional screens is related to opportunity since screens are relatively cheap and can also be shared between applications. Scalability questions relate to the usefulness of dedicating more screens to a visualization application when the screens are already available (*e.g*., [27]). Depending on the community, using multiple processor cores is considered as resource optimization or opportunity. We connect the resources and meaning of scalability in more detail in Section 3.2.
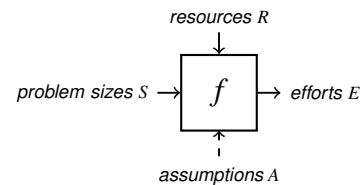

Fig. 1: Conceptual model with problem size variables S and resource variables R as input, assumptions A, and effort variables E as output to f.

3.1.3 Assumptions A

Assumptions define the validity bounds of the function f for the chosen research context and problem definition. More precisely, assumptions include the range of available resources, the range of reasonable values to expect for problem size, and the range of values considered acceptable or satisfactory for effort variables (*e.g.*, interaction rates perceived as interactive [14]).

3.1.4 Efforts E

The *effort* variables are properties describing the performance of the process. Effort variables can be measures of efficiency (*e.g.*, readability) or measures of cost (*e.g.*, computation time). For convenience, we consider that effort variables can always be expressed as a cost, where lower values are better. From a computational perspective, some examples of effort variables are computation time or frame rate. From a visual and user perspective, some examples are the ambiguity of a data representation (as opposed to its faithfulness), the interactivity of the system, the ease and speed of task completion, the number of insights, or visualization quality metrics. As in complexity theory, across all applications, effort can be defined as average, best, or worst case for a given set of problem size characteristics. Additionally, other aggregates across outcomes could be considered, such as standard deviation, for instance.

3.2 Meaning and Expression

Generally, a scalability issue is an inability of a technique or system to accommodate an increase in problem size, for the given resources. The inability is manifested by efforts that do not meet requirements, *e.g.*, inaccurate results, processing that takes too long, or a system failing to respond. A *scalable* system might address a scalability issue in different ways. In our model, the function f formalizes the relationship between the components relevant to scalability such that the *meaning* of scalability can be expressed as properties of f. We present three examples of meanings of *scalable* or *more scalable* in relation to f, illustrated in Figure 2: based on the shape of f, based on f passing a threshold, or on f demonstrating better performance on large problem sizes.

When scalability is the ability to sustainably handle increasingly large problem sizes with reasonable effort, it corresponds to concluding scalability from the *shape* of f. One example, illustrated in Figure 2a, is demonstrating that f (blue line) is linear with respect to increasing problem size at fixed resources. Another example is demonstrating that f is reciprocal with respect to increasing amount of resources at a fixed problem size, to show that an increase in problem size can be accommodated by more resources.

When scalability is the ability to handle problems of larger size than before, it corresponds to the function f_{new} sustaining the same amount of effort for a larger problem size than another f_{old}:

$$f_{\text{new}}(S_{\text{new}};R,A) = f_{\text{old}}(S_{\text{old}};R,A) = E_{\text{fixed}}$$

with $S_{\text{new}} > S_{\text{old}}$ and, ideally, identical resources and assumptions. Under this meaning, the threshold of interest, E_{fixed}, is typically an upper bound to the acceptable effort, for instance, the maximum reasonable latency. This case is illustrated in Figure 2b, with a threshold defined by the upper bound of the gray area.

When scalability is the ability to perform better for a range of problem sizes, it corresponds to the function f_{new} sustaining

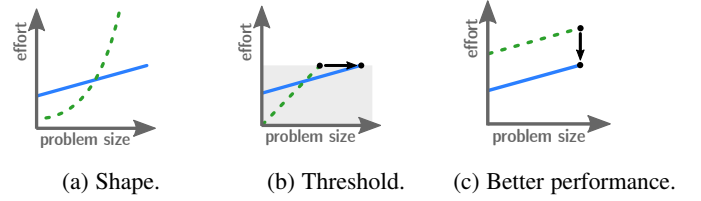

(a) Shape.   (b) Threshold.   (c) Better performance.

Fig. 2: Examples of effort functions, f_{new} in solid blue and f_{old} in dashed green, with f_{new} being more scalable than f_{old} according to three meanings of scalable.

lower effort than another f_{old} for all, or most, of the interval of S considered:

$$f_{\text{new}}(S;R,A) \leq f_{\text{old}}(S;R,A)$$

The function f_{new} may be better by a constant: $f_{\text{new}}(S;R,A) = f_{\text{old}}(S;R,A) - c$, $c > 0$ like the example of Figure 2c. The key difference with the meaning *threshold* is that there is not necessarily any concern with extending the range of supported problem sizes in that case.

Because the meanings of scalability are connected to the characteristics of f, the scalability claim in a paper is often supported by some level of description of the function f. We model this aspect by the *level of expression* of f, which is closely linked to the methodology employed. The function f may be described by an *explicit function* relating the set of problem size variables S and resources variables R to the set of effort variables E. An explicit function for f may be a model function or an approximation, for instance describing its asymptotic behavior with big O notation. Explicit functions are reported following a complexity analysis, mathematical proofs, or performance modeling. The function f may also be described by *sample points* that are measurements of effort variables for a sample of problem size variables. Samples points are reported through examples results, plots, or tables following a performance evaluation using synthetic data or datasets of varying problem size.

4 EXAMPLES OF MODEL INSTANTIATIONS

In this section, we present examples of instantiations of the model using different example papers structured in four stereotypical *scenarios* that are not meant to be comprehensive but rather didactic. The model, including the scalability meaning, corresponds to a scalability claim but scalability experiments/evaluation can also be described in terms of problem size, resources, effort, and assumptions.

4.1 Algorithm Scalability

Algorithm papers, including rendering papers, usually present a contribution for which instantiating the model is straightforward. In most cases, the problem size variables are clearly-defined properties of the input data, and the effort variables are fully explained and correspond to the computation cost in terms of execution time or memory. Because algorithm performance can be studied in a controlled manner, it is possible to evaluate their theoretical effort function. Still, most papers use an asymptotic model because the constants vary depending on hardware, configuration, etc.

Consider, for example, the problem of graph drawing that consists of optimizing the layout of the nodes of a node-link

diagram on the display. The approach based on solving a stress model requires computing the full matrix of all-pairs shortest paths between the graph nodes. This operation costs $O(|V|^2 \log(|V|) + |V||E|)$ in time and $O(|V|^2)$ in space for a graph with $|E|$ edges and $|V|$ nodes. For this problem, the problem size variables relate to the size of the input graph while the effort variables are the computing time and associated memory consumption. Khoury *et al.* [28] propose an approximation technique to solve this problem in quasilinear time and space and describe its effort function with detailed theoretical expressions of asymptotic complexity, together with experimental measures of computation time for a given implementation. An algorithm is usually considered scalable if it runs in linear time relative to the problem size.

Example: Drawing Large Graphs by Low-Rank Stress Majorization [28]

- -

Expression: explicit function (big O notation)
Meaning: shape (more scalable means more linear)

number of edges $|E|$ → f → execution time
number of nodes $|V|$ → → memory consumption

In some cases, the problem size does not relate to the input data but rather the size of the visual output. Falk and Weiskopf [29] consider the problem of rendering 3D vector fields using a texture-based representation. The computational cost of this problem is naturally cubic with the output image resolution. The proposed algorithm uses an image-oriented sampling approach and only computes the parts of the dataset that are represented on the final image. Consequently, this algorithm is mostly independent of the dataset size and predominantly governed by the output resolution and the number of samples. Performance measurements of a GPU implementation show that the rendering times are almost constant as the dataset size increases and scale linearly with the output image and the number of samples.

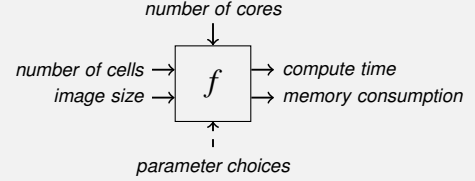## 4.2 Parallel Computing Scalability

This scenario is rooted in scaling experiments that are employed to understand the scalability of parallel computing implementations, typically in the context of HPC, cluster computing, or architectures with multiple cores or GPUs. These scenarios also occur in visualization research, mostly in large-data visualization with volume rendering or flow visualization. In this context, scalability is presented as a mapping to compute times with the effort measured for varying problem sizes and hardware resources (*i.e.*, compute nodes). In this scenario, one major concern is the optimization of computing resource usage, measured using *efficiency*, a metric that compares the gain in execution time (speedup) compared to the amount of additional resources made available to the system. The role of varying resources is extensively studied when looking at computational speedup in relationship to number of processors of compute nodes (*e.g.*, Gustafson's law [30] for weak scalability). This leads to assessing the explicit function of our model, related to resources varying together with problem size.

One such example is by Howison *et al.* [31]. This work assesses the scalability of volume rendering techniques using hybrid parallelism. They measure the problem size in terms of the size of the volume, here, the number of cells in a uniform 3D grid. They discuss that there might be additional data dependency due to the distribution of data values, but identify that there is no relevant impact from data-dependent early ray termination in their case. Effort is measured in memory consumption (MB) and

speedup of the compute times (*i.e.*, indirectly, the compute times). Scalability is primarily understood as the functional mapping to compute times (or speedups). To this end, they measure the effort for varying problem size and hardware resources (*i.e.*, compute nodes); in this sense, the functional mapping is represented by quite fine sampling of the function. In addition, they discuss further assumptions, in particular, parameter choices such as various types of block sizes used to distribute the compute work and the type of parallelism (hybrid parallelism vs. distributed memory only). Moreover, they also evaluate strong and weak scalability.

Example: Hybrid Parallelism for Volume Rendering on Large-, Multi-, and Many-Core Systems [31]

- -

Expression: sample points (fine sampling)
Meaning: shape (weak and weak-dataset scalability)

number of cores

number of cells → f → compute time
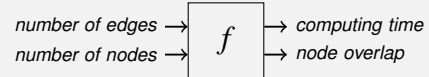image size → → memory consumption

parameter choices

4.3 Visual Scalability

One common scenario of scalability in visualization research targets the study of how the technique's or tool's visual performances are affected by the increase in size of the input data. In this context, scalability is presented as a mapping to readability primarily, and sometimes compute time as well, with the effort measured for varying problem size. These papers present new visualization techniques that can show a larger amount of data in a readable way through data aggregation, interaction, or smart visual encoding. However, they do not always use measurements for the effort function for the readability aspect, but rather discuss the limits of previous encodings, present the rationale behind the new one, and provide visual examples. When the computational aspect of scalability is discussed, it is often in relation to interaction latency. This scenario seems to be the most typical among visualization papers, at least it is the most commonly found in our survey (see Section 5).

Example: Structure-aware Fisheye Views for Efficient Large Graph Exploration [32]

- -

Expression: sample points (example datasets)
Meaning: better performance

number of edges → f → computing time
number of nodes → → node overlap

Wang *et al.* [32] tackle the visual clutter problem of large graph drawings using a focus+context interactive view. The problem size is the graph size. The main effort variables dependent on problem size are the number of overlapped node pairs which evaluates clutter, and the computing time which evaluates interactivity. The quality of the drawing as evaluated by two metrics (edge orientation offset and shape preservation) and task completion rate are also measured but not in relation to the graph size, *i.e.*, as concerns orthogonal to scalability. The authors do not claim that the technique is scalable but rather that their technique is made to mitigate scalability issues and that it outperforms other similar techniques that are compared in an evaluation using five datasets of different sizes.

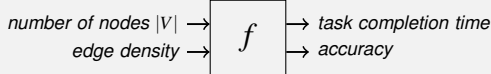## 4.4 Cognitive and Perceptual Scalability

This scenario covers studies from the psychophysics domain interested in user response to stimuli or studies to investigate the scalability of human perception and cognition when performing a visualization tasks. In this scenario, studies are close to an ideal controlled environment while being subject to the variability of humans. The effort variable is the user performance or cognitive load, measured in terms of objectively measurable metrics such as task completion time and accuracy. Problem sizes may be input size or task complexity. In this scenario, articles may present the issues under the terms *perceptual* or *cognitive scalability*, or may not mention *scalable* or *scalability* at all. In contrast to the visual scalability scenario, in this scenario, studies are not only interested in human limits regarding how visible elements are but also in the cognitive limitations.

Ghoniem *et al.* [33] compare the scalability of two representation techniques for graphs, adjacency matrix and node-link diagram, regarding their readability. The problem size variables are the number of nodes and the edge density, defined as $\sqrt{|E|/|V|^2}$ for a graph with $|E|$ edges and $|V|$ nodes. The effort variable is the readability measured by participants' response time and accuracy across seven tasks such as finding the most connected node or evaluating the edge density.

Example: A Comparison of the Readability of Graphs Using Node-Link and Matrix-Based Representations [33]

- -

Expression: sample points (example datasets)
Meaning: more scalable means faster completion rate on average

number of nodes $|V|$ $\rightarrow$ | f | $\rightarrow$ task completion time
edge density $\rightarrow$ | | $\rightarrow$ accuracy

Another example by Guiard *et al.* [19], investigates the scalability of Fitts' law for pointing tasks, relevant to selection and navigation in any pan-and-zoom environment. Fitts' law describes the empirical relationship between the user movement time and the pointing task difficulty, called index of difficulty $\text{ID} = \log_2(D/W + 1)$ where D is the target distance and W the target size. The time for pointing is $\text{MT} = a + b \times \text{ID}$. In the physical world, the ID is limited to about 10 bits. Using a zooming interface in a virtual world, the study shows that Fitts' law still applies beyond the limit. The effort variable here is the *throughput*, defined as the ratio between the ID and the movement time while the problem size is the ID. The study concludes that, for a higher range of IDs, the throughput of multiscale pointing is constant, so Fitts' law holds, limited only by human fatigue.

4.5 Non-stereotypical and Composite Scenarios

While these four scenarios are representative of a large part of the typical scalability concerns, we acknowledge that they are not exhaustive: some research work will lay at the intersection of scenarios and others may not exactly fit any scenario.

For systems handling many users, either interacting independently or in direct or indirect collaboration to complete a common task, scalability is understood as the capacity to accommodate for more users. In Glemarec *et al.* [34], the main challenge is to handle multiple user sessions at once: the problem size here is the number of users, while the effort variable is the system performance in terms of latency. In Deng *et al.* [35], the challenge is to collect a large training set of images with labels indicating the presence or absence of a set of objects of interest, by querying humans on

a crowdsourcing platform. Here, the problem size is made of the number of images and the number of objects of interest, which can both be large and users of the system are technically the resources available to complete the annotation task for a dataset. The effort variable is the number of human queries, which directly correlates with the financial cost of completing the annotation task for a dataset.

Similarly, systems using non-conventional display sizes (mobile displays or tiled displays) sometimes take interest in scalability with the number of composing screens/projectors or the overall display size. The dependent variables describing effort can be related to rendering characteristics like pixel error and compute time (in the form of frame rate) or user task completion time for testing human perceptual or cognitive limits in such configuration (*e.g.*, [11]).

4.6 Dependent Scalability Issues

Visualization system papers may include multiple aspects of scalability (visual, computing) or multiple components (querying component, rendering component) for which the scalability concerns differ and may depend on each other. We focus on this case to show how multiple scalability concerns can be modeled using multiple instances of the model, as illustrated in Figure 3.

A common pattern in large, interactive visual analytics systems is to render aggregated data and rely on a precomputation step to build a data structure supporting fast queries for interactive exploration. Some examples among many are data cubes used for multidimensional and spatiotemporal data exploration (*e.g.*, Nanocubes [36]), prefetched indexes for brushing and linking interactions [37], or custom in-memory hierarchical structures for detail-on-demand interactions [38]. In these systems, the scalability concern of the rendering step is trivially solved by representing pre-aggregated data, in the form of heatmaps, histograms, or clustered graphs for instance. Detail-on-demand interactions are based on the querying system that is, in turn, the focus of the scalability concern for increasingly large data. The use of a precomputed data structure to speedup queries trades performance improvement for query time at the cost of the precomputation step and the usage of additional memory to store it. Across systems, the assumptions and concerns differ regarding this preliminary step. In Nanocubes [36], the storage size of the data structure is a primary concern since the target is to store it on a laptop. However, their computing time is of lower concern, suggesting data is assumed to be static. In ASK-Graph [38], no scalability concern is listed for the precomputation step, however, the data structure computation is parameterized by the available resources: available RAM and maximum number of edges that can be processed in a few seconds.

Scalability concerns can also vary as research advances. For instance, for spatial and multidimensional data cubes, Hashed-Cubes [39] is presented as an improvement over the state of the art regarding storage size and query time. In addition, they also present building time as an important factor, unlike previous work, and mention supporting dynamic data as future work.

This illustrates that the criteria of scalability can be different across the components of a system (precomputing, querying, rendering), sometimes functioning together. They also depend on the context of application, *e.g.*, static vs. dynamic data, or client resource requirements. In turn, the amount of space dedicated to these different levels of scalability concerns vary in articles.
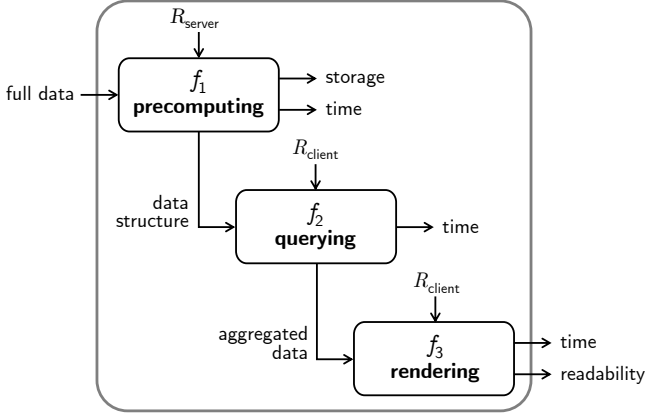
Fig. 3: Dependent scalability issues as multiple instances of the model for a common pattern in large, interactive visual analytics systems: aggregate visualization using a precomputed data structure to enable fast interactions. *Rendering* the visualization has no scalability concern since it is ensured, by design, to only handle aggregated data. The focus is on scaling the *querying* step, while the scalability concern for the *precomputing* step varies across applications.

4.7 Comparison with Related Work

Related work mentioned factors and measures; they become our problem size, resources, and efforts. The distinction between factors and resources is clarified in our model. The *nuisance variables* can belong to problem size or assumptions, depending on their nature. Regarding multi-criteria optimization advocated by Duboc *et al.* [9], our efforts could be expressed with a combination of criteria if desired. In addition, our model also specifies different meanings and expressions of scalability that are usually implicit in each community or domain but can be hard to understand when the readers are from very different backgrounds. Finally, visualization and visual analytics are evolving domains in search of better evaluation methods. Therefore, the meaning and expression of scalability will evolve and require clarification. We that believe our model can be a first step to encourage the diversification of scalability definitions and help support the development of scalability evaluation methods, especially for research related to the *Visual Scalability* and *Cognitive and Perceptual Scalability* scenarios.

5 LITERATURE ANALYSIS

To relate our model and scenarios to the current state of visualization research, we conducted a structured and systematic literature analysis based on a coding scheme describing the different components of the scalability model. We carried out iterative coding rounds to form and refine a *coding scheme* describing the meaning and reasons supporting scalability claims in visualization. Methodologically, we followed a manual coding process that originally stems from qualitative research and allows for the systematic and structured analysis of literature under ill-defined tasks that necessitate human coders [40]. Such approaches have been frequently used in visualization research, for instance, to characterize evaluation methods [13], [41], interactive dimensionality reduction [42], and ensemble visualization [43]. In the following, we describe our methodology, analysis process, and quantitative and qualitative results.
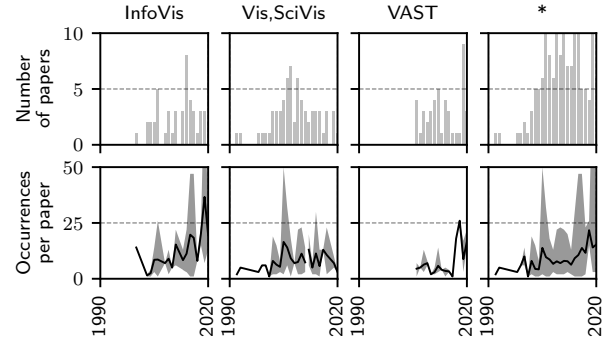

Fig. 4: Number of papers and full-text occurrences (band: min-max, line: mean) of the terms *scalability* and *scalable* in the sample of 127 papers, per year. The first three columns show charts per conference, the last column shows charts for the whole.
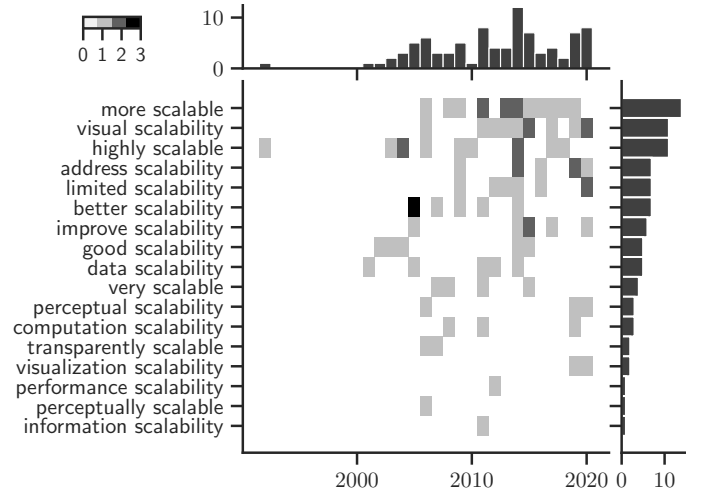

Fig. 5: Top-17 occurring expressions including the terms scalability or scalable with number of papers from the sample of 127 papers that use them in their full-text content across years and all papers.

5.1 Literature Sample

The selection of papers included in our study is based on the `vispubdata` dataset [44], containing 3394 papers from the IEEE Visualization (VIS) conferences from 1990 to 2020 (conference and journal articles as well as panel or poster papers). This initial set of papers was filtered automatically to select papers including the prefix *scalab-* at least once in their abstract, title, or keywords in order to capture papers for which scalability was a main concern. This process generated a set of 157 relevant papers. After removing 30 papers because they mentioned scalability but did not discuss it, we were left with a corpus of 127 papers: 35 conference, 87 journal, and 5 others articles, among which 6 were of length 5 pages or fewer (full list available at `https://osf.io/xrvu7/`). Figure 4 gives an overview of the frequency of how often the respective terms were used in these papers, organized by year and conference. Figure 5 shows the frequency of the top-17 occurring expressions related to scalability. This list gives a picture of the different aspects and meaning of scalable and scalability.

5.2 Coding

The coding process started with an open-coding phase during which four co-authors of the paper, who developed the model, reviewed

TABLE 1: Summary counts for the corpus. Left: Conference counts for the collected and coded papers. Right: Counts for papers excluded, with or without consensus coding, and edge cases.

Venue	Coll.	Coded	Set	Count	Edge Cases
VAST	44	37	Excluded	30	-
Vis	47	36	Coded	127	14
SciVis	19	16	Consensus	21	5
InfoVis	47	38	No consensus	106	9
Total	157	127	Total	157	14

Fig. 6: Distribution of codes for the 127 coded papers.

32 seed papers to identify existing variables for the components of our conceptual model for scalability. This phase allowed us to get an overview of the selected literature and led to the development of categories of fixed codes constituting a coding scheme. After multiple iterations, the coding scheme was tested by the four coders on a small sample of 12 papers. The coding scheme is structured by the two objectives of our literature review, which are to describe (1) **what** the scalability claim is about in a paper, and (2) **how** scalability is presented in the paper. The coding scheme has four categories of codes for the former, and two for the latter:

Input covers the problem sizes and resources of the effort function associated to the paper's scalability claim, *i.e.*, they describe what varying parameters are considered, jointly to simplify the coding book. Inputs can be multiple: *Data Size*, *Data Characteristic*, *Compute Nodes*, *Display Resolution/Units*, *User/User Sessions*.

Output covers the outputs of the effort function, *i.e.*, the costs or dependent parameters considered in the scalability claim. Outputs can be multiple and may be qualitative (not measured): *Compute Time*, *Memory Consumption*, *User Performance*, *Error/Quality*, *Clutter/Readability*.

Meaning covers three different, mutually exclusive, meanings for *being scalable*, defined in relation with our model's effort function. *Function Shape* characterizes scalability based on the shape of the function (constant, linear, bounded). *Extend Domain* defines being scalable as being able to handle problems of larger size than before. *Better Performance* defines scalable as the exhibition of better performance compared to another technique, *i.e.*, having a lower effort for the same inputs.

Scenario covers the four scenarios from Section 4: *Algorithm & Rendering Scalability* (ALGO), *Parallel Computing Scalability* (PARA), *Visual Scalability* (VISU), *Cognitive & Perceptual Scalability* (PERC). We hypothesize that these scenarios will be able to capture/separate the different definitions of scalability and practices of visualization subcommunities.

Expression describes how explicit the effort function is in the paper: *Model Function* and *Asymptotic Function* for function expressions, *Plots/Tables* and *Few Samples* for sample points.

Reasons gives possible arguments exposed, or evaluations conducted to support the presented expression of the effort function and associate meaning of the scalability claim: *Unspecified*, *Didactic/Argumentative*, *Theoretical Validation*, *Experimental Validation*, *Case Study/Examples*.

During a second phase, the two other co-authors, who did not participate in the preliminary phase leading to the coding scheme, coded the complete set of papers, to validate the coding scheme and quantitative insights from the literature. These new coders were first introduced to the mod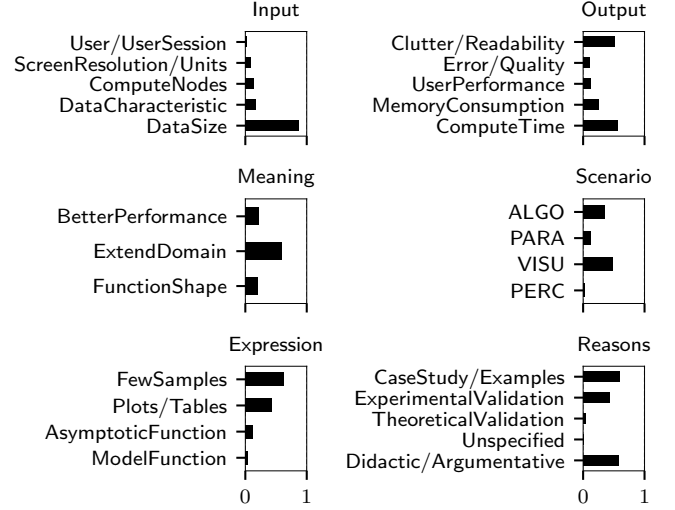el and trained on four examples. Some aspects initially part of the coding scheme (number of comparisons with other techniques, assumptions) were removed for simplification and *User/User Session* added to input codes. The coding process progressed in batches, identical for both coders, starting with the 12 papers previously coded for calibration. After each batch, the coders reviewed low-agreement papers to obtain a consensus coding. During that process, they excluded 30 papers for lack of details about scalability and identified 14 "edge case" papers that barely fit into the coding scheme. Out of the 127 papers from the coded corpus, 21 were assigned a consensus coding and 106 were assigned the average of the two coders' coding (see Table 1). The inter-coder agreement was .72 for the 127 papers when including the initial low-agreement codes of the 21 papers that benefited from a consensus coding, and .76 for the 106 papers that did not (Bennett, Alpert and Goldstein's S [45] with Jaccard distance). Using the interpretations of Cohen's kappa, these scores denote substantial agreement (between .61 and .80).

5.3 Quantitative Results

An overview of the coding results for our corpus is shown in Figure 6. Most frequently, scalability claims are related to *Data Size* (Input), and concerned with *Compute Times* and/or *Clutter/Readability* (Output). The most typical scalability claim is *Extend Domain* (Meaning). The most frequently represented scenarios are *Algorithm & Rendering Scalability* and *Visual Scalability* (Scenario). The distribution of the expression codes follows their level of care for strictness with the majority of papers reporting at least some aspect of scalability with singular examples (*Few Samples*), whereas very few described the effort function with the precision of a *Model Function* (Expression). Although linked to the common big *O* notation for describing algorithm complexity, the *Asymptotic Function* code remains rarely reported. The most commonly found (non-exclusive) reasons to support scalability claims are *Didactic/Argumentative*, *Case Study/Examples*, and *Experimental Validation*.

To highlight the relationship between the codes of the coding scheme and identify those that are consistently used together, we look at the correlation between codes across the corpus. Figure 7 shows the clustered correlation matrix of the codes for the 127 papers. We find the most common coded scenario, VISU
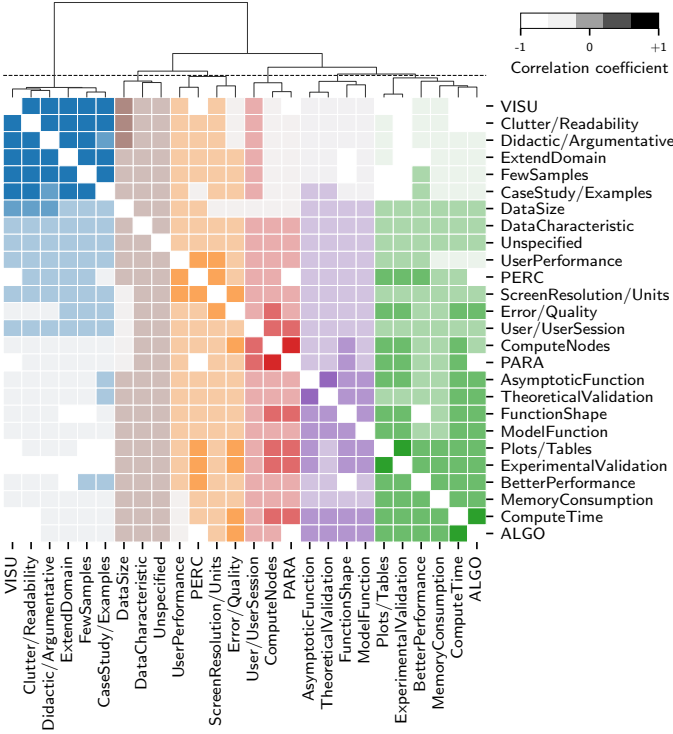
Fig. 7: Correlation between codes as a clustered heatmap. Rows and columns are ordered identically, following the order of the dendrogram leaves. Cell hue indicates cluster membership following the six clusters from the represented dendrogram cut (hierarchical clustering: Ward linkage criteria), while cell intensity encodes correlation coefficient between codes (Pearson coefficient).

(*Visual Scalability*), is linked to the codes *Clutter/Readability*, *Didactic/Argumentative*, *Case Study/Examples*, *Few Samples* (in blue), which describe almost all aspects of the typical visual scalability scenario. The green cluster highlights the codes that are related to the ALGO scenario, such as *Model Function* and *Function Shape*. The orange cluster, less defined, includes the PERC scenario with some related codes such as *User Performance* (Output). We also find several, unsurprising, pairs of highly correlated codes such as *Compute Nodes* and PARA (red), or *Asymptotic Function* and *Theoretical Validation* (purple).

Figure 8 presents an overview of the papers. At the top, papers are represented as a UMAP [46] 2D projection of their coding and colored by coded scenario. The UMAP projection reveals a clear split between two clusters. At the bottom, the average coding per scenario and per cluster is represented as a heatmap. The cluster on the right resembles papers approaching scalability primarily from a visual angle, as characterized in the VISU scenario (blue), with *Clutter/Readability* as an effort variable. The papers from the other scenarios in this cluster are likely those that discuss aspects of scalability that are primarily associated with the blue scenario (*i.e.*, *Didactic/Argumentative*, *Case Study/Examples*, *Few Samples*). For example, three of the red papers are related to tiled-displays (screen or projectors). The cluster on the left covers papers approaching scalability primarily from a compute angle, as characterized in the ALGO (green) and PARA (red) scenarios, with *Compute Time* as an effort variable. The papers from the other scenarios in this cluster are those that discuss aspects of scalability that more often associated with the green scenarios (*i.e.*, *Plots/Table*, *Experimental*
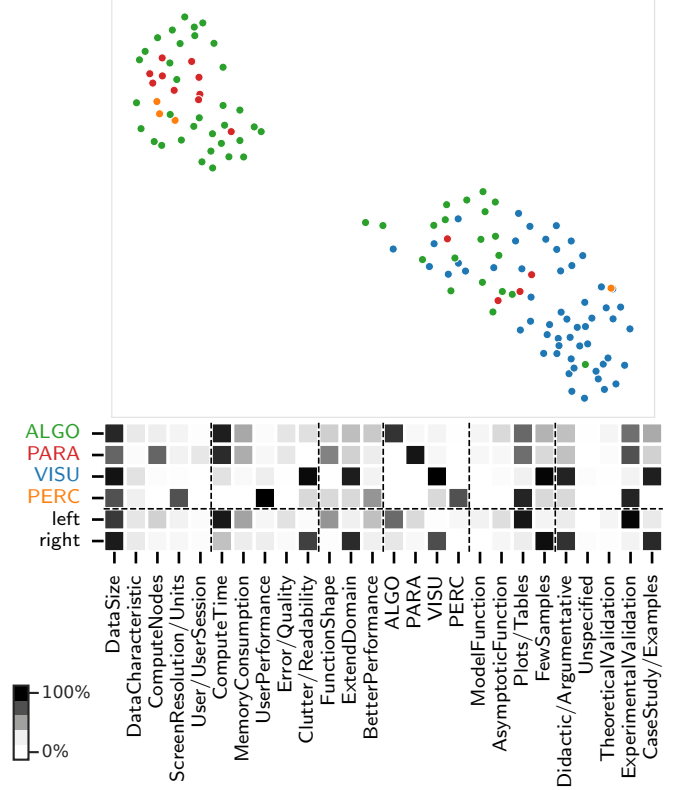

Fig. 8: Overview of the coded corpus. Top: UMAP embedding of the 127 papers of the corpus based on their codes (excluding the *Scenario*) and colored by scenario. Bottom: distribution of papers codes per coded scenario and per projection cluster (left/right). Cell intensity indicates the proportion of papers from the row group that have the column code: black indicates all, white none.

Validation). Overall, this overview shows that multiple aspects of scalability coexist in the community, and even within our four stereotypical scenarios. It also hints to categorizations of scalability papers finer than, or different from, our scenarios.

5.4 Results per Scenario

The four scenarios were devised after reviewing part of the corpus papers, being aware that some papers will differ from stereotypical cases. In some cases, the scenario in which the scalability claim was demonstrated is different from the scenario of the overall paper. For example, some of the papers that belong to the ALGO scenario demonstrated the scalability using parallel implementations (*i.e.*, PARA scenario) or by conducting a user study (*i.e.*, PERC scenario). During the coding phase, coders assigned a single scenario per paper and picked the one that matched more closely the overall paper context. While the majority of papers were successfully coded into one of the four stereotypical scenarios, the coders also found recurrent types of papers within each scenario.

In the ALGO scenario, the typical paper presented rendering algorithms for scientific data describing biomedical, spatial, or physical phenomena. As shown by their average coding in Figure 8, these papers typically have *Data Size* and/or *Compute Nodes* as input, and *Compute Time* and/or *Memory Consumption* as output. To measure the scalability of the proposed technique or method, they usually conduct an extensive *Experimental Validation* by varying the input parameters and the results is often communicated

in forms of *Plots/Tables*. Another type of papers from this scenario address multi-projector displays (6 papers). In these papers, scalability applies to a new calibration method and is relative to an increase in the number of projectors, which challenges the quality of the final picture. Here, the encoded input corresponds to *Screen Resolution/Units*, while the encoded output corresponds to *Error/Quality* and, in some cases, *Compute Time*. In these papers, the scalability claim is often demonstrated by showing few pictures of the final tiled display (*i.e.*, *Few Samples*).

The typical paper in the VISU scenario introduced a new visualization technique, or a novel visual analytics tool that aims at solving specific analysis tasks for targeted domain users. As shown by the average coding for VISU papers in Figure 8, the typical coded input and output are *Data Size* and *Clutter/Readability*, respectively. The typical scalability claim is demonstrated by a *Case Study* combined with *Didactic/Argumentative* discussion based on few and sometimes only one dataset and has the meaning of *Extended Domain*. About half of these papers address the scalability problem by employing summarization, aggregation, or sampling strategies on the data layer, the presentation layer, or both and often in combination with interaction techniques (27 papers). The rest address the scalability problem with new techniques (*e.g.*, layout, interaction) or new visual analytics systems combining existing visualization methods, but without aggregation.

Papers in the PARA scenario typically describe a new parallel implementation of an existing algorithm, or a new system using a parallel architecture that can scale up to numerous *User Sessions* without introducing undue delay (*i.e.*, load scalability).

Finally, only 5 papers matched the PERC scenario. These papers are generally concerned with measuring *User Performance* relative to varying *Screen Resolution/Units* and *DataSize*, one example being the study of perceptual scalability on large tiled-display walls. While no single scalability meaning emerges, all papers present *Experimental Validation* with results communicated in the form of *Plots/Tables*.

5.5 Edge Cases

Throughout the coding process, the coders marked 14 papers as difficult to code using the coding scheme. These papers were additionally open coded for the reasons why they did not fit, to evaluate the limitations of the coding scheme and the conceptual model. After an open-discussion session with all co-authors, we identified four different, non-exclusive, reasons of difficulty:

1) **Require inference:** 7 papers presented a scalability claim (often in the beginning) without establishing a clear link between scalability and the results in the rest of the paper. A common pattern was a switch in the terminology used (*e.g.*, using scalability first and then performance).

2) **Use of a scalable related-work component:** 2 papers presented visualization systems relying on a subcomponent or a supporting system said to be scalable. They discussed the scalability of another system, without necessarily relating it to the scalability of their contribution.

3) **Limited scalability:** 1 paper did not claim scalability but rather discussed scalability issues and limitations of their work. While this is a good scientific practice, it was difficult to code, particularly the *Meaning* code.

4) **Other meanings:** 2 papers used the word scalability to describe concepts we believe could be best described by other words, *e.g.*, adaptability, automation, flexibility.

In the latter case, we believe authors could use our model to describe the discussed aspects of scalability and thus, convey a clearer and more meaningful message. While these edge cases did not challenge the coherence of the model, they raised questions about the difference between scalability and other system properties like flexibility.

5.6 Summary and Discussion

While the coder agreement confirms the validity and applicability of our model to visualization research, we also observed that the coding scheme fell short of precisely capturing the scalability concern or claim of some papers in our corpus. One reason is that several notions of scalability can coexist in the same paper, for instance connected to different components. For these cases, the paper coding ended up describing multiple aspects as a single effort function, but also grouping different types of reasons together even when they each corresponded to a single effort variable. Although not covered by our coding process, the multiple scalability considerations in a paper could be coded more precisely as different coding instances. Another reason is the lack of consistency in the terminology used that made it difficult in some cases to connect the general scalability discussion to detailed evaluations and results. In other cases, it was not clear if the authors wanted to communicate an improved scalability or scalability limitation through their evaluation. Our model is meant to address most of these cases, to help authors clarify and expose their claims.

The literature analysis gives an overview of the types of scalability discussed in the corpus and how scalability claims are presented and supported in the corpus of papers. The most frequently represented scenarios are *Visual Scalability* and *Algorithm & Rendering Scalability* with *Computation Time* and *Clutter/Readability* being the two most common types of effort considered. The meanings of scalable are various even within the same scenario category, the most common being the ability to supporting larger problem sizes than before (*Extend Domain*).

We acknowledge that this overview depends on the balance of topics in the venues chosen for our corpus of papers. Our corpus may be biased toward less of the traditional scalability papers, from the computer graphics community for example, that may be presented at IEEE VIS but originally published in TVCG and not covered by our pool of papers. However, we believe that the IEEE Visualization (VIS) conference publications are, at least for the last decade, representative of the publications in the domain. Moreover, our filtering process also comes with some limitations: similar to our corpus including papers using *scalable* to refer to concepts different to our interest (false positives), some other papers discussing scalability issues under different terms or only in the body of the paper could be missing (false negatives). This could have affected papers from the *Cognitive & Perceptual* scenario for instance, as they represented only a small portion of our corpus. Our filtering process shows that roughly 5% of the papers were concerned with scalability at the IEEE VIS conferences. To provide some context, we filtered the list of publications from other venues relevant to the visualization domain using the same criteria. We report the numbers and portion of collected publications for these other corpora of papers in Table 2. Around 5% or fewer papers are recovered for venues with a broad scope (VIS, EuroVis, TVCG), and around 20% for the EGPGV and LDAV symposia, which are focused on parallel/large-scale graphics and visualization. This is not surprising since scalability

is a major topic in parallel/large-scale visualization, and less so in other conferences with a broader scope. We can anticipate that the proportion of papers discussing scalability in conferences with a broader scope will raise as other visualization subcommunities also develop definitions and methodology to evaluate scalability.

TABLE 2: Number of publications collected for different venues, using the same filtering approach as used for our corpus.

Venue	Collected
IEEE Visualization (VIS) conf. (1990–2020)	157/3394 (5%)
EuroVis (2000–2020)	44/1263 (4%)
IEEE TVCG (1995–2020)	209/3913 (5%)
CGF (1982–2020)	118/4447 (3%)
EGPGV (2002–2020)	39/212 (18%)
LDAV (2011–2020)	43/191 (23%)

6 RECOMMENDATIONS AND EXAMPLES

According to our review, the term scalability is used with multiple meanings and the claims of scalability are sometimes difficult to interpret. The different visualization subcommunities may have different traditions related to scalability. Most of them borrow methods from other computer science fields such as algorithms and databases. The HPC visualization community is familiar with scalability issues but, when attempting to publish work spanning across subcommunities should make sure the well-known HPC/visualization issues are understood by the others. Similarly, scalability about human issues may be understood by communities connected to HCI or psychology but not always by *e.g.*, the computer-graphics community within visualization. Therefore, we see the need for a clear communication between the wide range of fields and subcommunities that play a role in visualization research.

In the following, we provide a set of recommendations for researchers without a scalability tradition and authors targeting the diverse crowd spanning multiple visualization communities, together with example papers from our corpus. Clarifying scalability claims will benefit the research process, the readers of the resulting papers, their reviewers, and eventually the visualization research community as a whole. It could even guide future research to cover scalability aspects more thoroughly.

For Researchers

We are convinced that considering scalability right from the start, and all the way through, is highly beneficial to a research project. We envision that doing so is similar to the thorough considerations that researchers already put today into other evaluation questions [13], [41], [47].

Incorporate scalability early on: Assessing scalability needs planning, and it cannot be done well at a late stage of the work. Choose a scalability goal early and decide how to support your scalability claims. In particular, incorporate scalability considerations in the evaluation of your work.

Sharpen the expression and meaning of scalability: Only   a few visualization papers try to model or measure a scalability function, or mention the asymptotic behavior. Several papers just report a few measurements. Higher-level characterizations are valuable because they provide a more informative view on scalability. Boosting the description of the scalability

expression and meaning has to be included early in the research process.

For Authors

The following recommendations can help authors improve the presentation of scalability in papers, especially when unfamiliar with scalability issues. Most of these recommendations are a follow-up of the literature review and the difficulty found when parsing the edge case papers.

Clarify scalability: Stating or explaining the meaning of scalability in the context of your work can help resolve ambiguity. The explanation could be kept short as long as the meaning becomes clear. This can be done by relating your work to existing and well-established scenarios (*e.g.*, [48]).

Consider our conceptual model: We expect that many explanations of scalability can be simplified by using our model. In particular, by describing *problem size*, *effort* considered, *resources*, and *assumptions*.

Take similar papers as examples: Our structured analysis facilitates finding papers that target certain aspects of scalability and, thus, are inspiring presentations for your own work.

Discuss limits and assumptions: Many techniques are scalable up to some limit and/or under certain assumptions, and these should be explicitly mentioned. Similarly, complex systems tend to be hard to evaluate—the evaluation is often restricted to picking just a few measurements. The underlying assumption for the evaluation should also be documented (see [49], [50]).

Do not overload terms: Do not use the words "scalability" or "scalable" as synonyms for "efficient," "fast," "good performance," "faster than baseline," etc. Refrain from using them as buzzwords and use terms consistently to avoid ambiguity. The paper by Abello *et al.* [38] serves an example where different concepts such as scalability, flexibility, and usability are distinguished.

Match the description and importance of the claim: Supporting a strong claim of scalability requires explanations, sometimes equations, tables, or figures. Weaker scalability claims can get by with shorter explanations and less supporting evidence. Choose the right balance according to the importance of scalability in your paper (*e.g.*, [51]).

Consider more than one scalability claim: A paper may make multiple scalability claims, or present different scalability characteristics that are then combined into an overall scalability assessment, as explained in Section 4.6. Document scalability for each of the claims, considering the above recommendations (*e.g.*, [51], [52], [53]).

Provide a nil-report: If a solution to a concrete problem is proposed but does not scale, report it—as an element in a fair assessment of your work. It may encourage others to improve upon your solution (*e.g.*, [54], [55]).

For Reviewers

More explicitly considering scalability can also be beneficial when reviewing papers.

Be specific with required revisions: When asking for more information about scalability, be specific. The above recommendations for paper authors can be used to clarify expectations.

For the Research Community

Finally, we also see the role of the visualization research community as a whole.

Fostering interdisciplinary communication: Many communities of visualization, but even more importantly, outside of visualization, already have well-understood interpretations and meanings of scalability. However, these often differ across subfields of computer science. Therefore, an explicit description of the type of scalability implied can help communicate research outside the community. This is particularly important for research that will concern such an outside audience.

Include scalability in best practices: We should strive to improve the way we discuss scalability in our papers, to better compare our approaches and make progress. To this end, existing best practices for conducting and reporting research should be further extended to cover the relevant scalability considerations.

Applying the Model: Coding Examples

In Table 3, we provide examples of how we coded the paper corpus. We do so by quoting selected excerpts from the selected papers and show how they map to our coding scheme. These excerpts are good examples of ways to mention the main model components in a research paper, which we recommend authors to do to present definitions accessible for all visualization research domains.

7 CHALLENGES AND FUTURE DIRECTIONS

Clarifying scalability in visualization goes beyond improving the current state of communicating research results, it also highlights open issues for the visualization research community.

7.1 Scalability and Evaluation

Characterizing scalability according to our model is tightly linked to the more general problem of evaluating visualization. There is an ongoing discussion in the visualization community about appropriate ways to perform evaluation [13], [41], even a full workshop dedicated to the topic since 2006 (BELIV).

Evaluation is especially hard when we want to assess human-related aspects. This directly relates to the problem of measuring the effort in our conceptual model. With experiments involving human participants, one can arrive at samples of the measured outputs. However, some performance measurements are notoriously hard to characterize or measure, such as readability and understanding.

Therefore, it is interesting to use existing or develop new proxies for measurements such as memorability, readability, or discriminability [66]. Even harder is the development of computational models for evaluation, or tools to facilitate the evaluation of techniques and systems such as EvalBench [67] and Touchstone [68]. They would be extremely useful to support faster evaluation and enrich the scalability expression, but they can be very difficult to do in general; they apply to visualization or interaction techniques but not to systems. However, fundamental evaluation issues are not restricted to human-oriented studies. For example, it is already hard to characterize all relevant input parameters and assumptions for a complex technical system such as a multi-GPU cluster running an advanced volume rendering system. Control of input parameters is also related to the issue of external validity of experiments, for example, for real-world data (uncontrolled) vs. synthetic data

(controlled, but not realistic). One promising solution could be the extension of the approach of generative data models [69]. This is also linked to developing additional quantification methods for visualization [70]. In summary, we see the need for further developments in evaluation methods to improve the characterization of scalability.

7.2 Reproducibility, Comparison, and Benchmarking

Addressing scalability in a visualization article is important not only for the paper but also for the community, to build knowledge about the comparative scalability of related research work. We are interested in monitoring the progress in research on scalability over the years, but we still lack consistency in scalability reports and meanings. Reproducibility and replicability [71] help tremendously for these comparisons and the community should incentivize visualization research to increase the number of reproducible articles. Otherwise, similar articles may decide to report about their scalability with very different choices, defeating or lowering the purpose of reporting scalability. We do not argue for a total alignment of research methods but rather to a convergence toward a consensual set of effort measures, methods to collect them, and to report them, with flexibility in reporting extra measures or using alternative, hopefully, improved methods.

With sufficient maturity, these comparisons should become benchmarks. Proper benchmarking and comparison between visualization contributions are important to drive research in our community [72], [73]. While these are relevant in general, they lead to specific challenges for scalability research because they need to include variations in problem size and properly control the assumptions. There are a number of (implicit) benchmarks in our community, for example, including the Contests at IEEE InfoVis, VAST, SciVis, and the Graph Drawing conference, but these usually rarely consider scaling the problem size. A related problem is the comparison of the resulting efforts. While this is doable for individual measures (for example, comparing overall compute times), it might be harder for cases that come with multiple measurements (for example, for different components of a complex system; or when assessing several facets such as speed, accuracy, readability, etc.). This leads to considering multiple efforts or multi-criteria efforts, mentioned by Duboc *et al.* [9] but new in visualization.

7.3 Role of Scalability in Future Research

A quite fundamental question is the extent to which we—as a community—should consider scalability in future research. Should a majority of papers address scalability as a relevant aspect to characterize visualization contributions? Or is it more appropriate to restrict the discussion of scalability to research in core subfields?

The SciVis community has been involved for many years with HPC where scalability is central. In-situ visualization [74] is strongly concerned with scalability but in the sense of not interfering with HPC code while providing useful visualizations and sometimes steering. The VAST and InfoVis communities have no identified subfield addressing scalability explicitly, except progressive visual analytics [75].

The community should keep track of its advancements in scalability and report its progress in a more structured way. Exposing progress in scalability would be an incentive and a useful asset for our community.

TABLE 3: Coding examples using excerpts from our corpus to illustrate each component of our model.

Input → Outputs	
Screen resolution/units→User performance	*"We evaluate the scalability limits of large, high-resolution, immersive displays [. . .] Our main metrics concerned user performance, specifically elapsed time." [56]*
Data size, Compute nodes	*"[...] is scalable with respect to both large data sets as well as future graphics hardware." [57]*
User/User session, Compute nodes→Compute time	*"Our framework should support many remote user sessions simultaneously. The performance should scale under an increased rendering load as hardware resources are added. [. . .] our performance metric is the turnaround time." [58]*
Screen resolution/units→Compute time, Error/Quality	*"Our approach efficiently scales to projector arrays of arbitrary size without sacrificing alignment accuracy." [59]*
Data size, Data characteristics→Compute time	*"[...] we evaluate our system to assess scalability in data size and data dimension." [60]*
Data size→Clutter/Readability	*"After reordering, adjacent tasks are aggregated to form a single task block, which will significantly reduce the clutter in visualization and actually make the schedule visualization more scalable." [61]*
Meaning	
Better performance	*"[...] we find that Protovis provides up to 20×higher frame rates than prefuse [...]" [62]*
Extend domain	*"Our study was designed to investigate the perceptual scalability of node-link diagrams for graph connectivity tasks, identifying the graph complexity and size beyond which they cease to be useful for such tasks." [63]*
Function shape	*"[...] the frame times show a favorable sublinear scaling instead of linear scaling as the number of render sessions increase." [58]*
Expression & Reasons	
Plots/Table	*"As shown in Table 1, Protovis consistently has frame rates an order of magnitude higher, up to 20 times faster for large graphs." [62]*
Asymptotic function	*"Overall bound: $O(n\log(n) + N\alpha(N) + g \times NS)$." [50]*
Case study	*"We demonstrate a case study with 276 samples which is considered a large study of mRNA-seq data by current standards [...]" [64]*
Experimental validation	*"[...] we describe additional experiments to evaluate the scalability of Dis-Function in a controlled manner. Specifically, we examine the performance of Dis-Function as the dataset grows in size (in terms of number of rows) and in complexity (in number of dimensions) independently." [60]*
Asymptotic function, Theoretical validation	*"These fields can be computed in linear time on the GPU and queried in constant time. Therefore, the complexity of the algorithm is reduced from quadratic to linear." [65]*

7.4 New Scenarios

With our quite flexible conceptual model, we want to keep scalability open for further interpretation and scenarios. In particular, we want to specifically avoid boxing-in any future research; that is why we did not aim to define scalability but instead, we provide a generic framework. We envision that new scenarios, or even a new set of scenarios, and best practices will be developed by our research community. This could even include new strategies to improve scalability, for example, new paradigms in computing, alternative ways to deal with various trade-offs, or novel representations.

One challenging problem relates to summarization techniques in visualization, such as Scagnostics [76] or "Accordion Drawing" [77], designed to remain readable by selectively showing specific aspects of the data. This approach reduces the problem size drastically, using clustering, sampling, or aggregation techniques, to address the display resolution bottleneck for instance. We think this is a specific case of scalability claim, as it corresponds to a trade-off between the level of information shown and the problem size. More work is needed to properly characterize the scalability of these multi-aspect techniques; it will require improving how assumptions are reported in addition to identifying the meaningful measures.

8 CONCLUSION

We presented a conceptual input–output model that allows us to characterize different scenarios of scalability in visualization research. We used the model as a lens to systematically analyze existing research on scalability in visualization, derive recommendations for communicating scalability across different

subcommunities, and highlight the open issues for scalability in the community. We hope that our work will help others, especially in the information visualization and HCI communities, to more easily and precisely characterize their scalability claims, and also to inspire them to conduct more research into scalability definition and associated evaluation methods. After all, the increase of data will most likely continue and we, as a community, will need to keep pace by ensuring that our contributions "scale" along.

REFERENCES

[1] C. R. Johnson, "Top scientific visualization research problems," *IEEE Comput. Graph. Appl.*, vol. 24, no. 4, pp. 13–17, 2004.
[2] P. C. Wong, H.-W. Shen, C. R. Johnson, C. Chen, and R. B. Ross, "The top 10 challenges in extreme-scale visual analytics," *IEEE Comput. Graph. Appl.*, vol. 32, no. 4, pp. 63–67, 2012.
[3] J. J. Thomas and K. A. Cook, Eds., *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. National Visualization and Analytics Ctr, 2005.

[4] D. A. Keim, J. Kohlhammer, G. P. Ellis, and F. Mansmann, *Mastering the Information Age – Solving Problems with Visual Analytics*. Eurographics Association, 2010.

[5] C. B. Weinstock and J. B. Goodenough, "On system scalability," Carnegie Mellon University, Software Engineering Inst., Tech. Rep. CMU/SEI-2006-TN-012, 2006.

[6] A. B. Bondi, "Characteristics of scalability and their impact on performance," in *Proc. Workshop. Software Perf.* ACM, 2000, pp. 195–203.

[7] M. M. Michael, J. E. Moreira, D. Shiloach, and R. W. Wisniewski, "Scale-up x scale-out: A case study using Nutch/Lucene," in *Proc. IEEE Int. Parall. Distrib. Processing Symp.* IEEE, 2007.

[8] M. D. Hill, "What is scalability?" *ACM SIGARCH Comput. Arch. News*, vol. 18, no. 4, pp. 18–21, 1990.

[9] L. Duboc, D. S. Rosenblum, and T. Wicks, "A framework for modelling and analysis of software systems scalability," in *Proc. ACM/IEEE Int. Conf. Software Eng.* ACM, 2006, pp. 949–952.

[10] G. G. Robertson, D. S. Ebert, S. G. Eick, D. A. Keim, and K. Joy, "Scale and complexity in visual analytics," *Inf. Vis.*, vol. 8, no. 4, pp. 247–253, 2009.

[11] B. Yost and C. North, "The perceptual scalability of visualization," *IEEE Trans. Vis. Comput. Graphics*, vol. 12, no. 5, pp. 837–844, 2006.

[12] S. G. Eick and A. F. Karr, "Visual scalability," *J. Comput. Graph. Stat.*, vol. 11, no. 1, pp. 22–43, 2002.

[13] H. Lam, E. Bertini, P. Isenberg, C. Plaisant, and S. Carpendale, "Empirical studies in information visualization: Seven scenarios," *IEEE Trans. Vis. Comput. Graphics*, vol. 18, no. 9, pp. 1520–1536, 2012.

[14] S. K. Card, T. P. Moran, and A. Newell, *The Psychology of Human-Computer Interaction*. Erlbaum, 1983.

[15] D. A. Szafir, S. Haroz, M. Gleicher, and S. Franconeri, "Four types of ensemble coding in data visualizations," *J. Vision*, vol. 16, no. 5, pp. 11–11, 2016.

[16] A. Treisman, "Preattentive processing in vision," *Comput. Vision Graph. Image Processing*, vol. 31, no. 2, pp. 156–177, 1985.

[17] C. G. Healey and J. T. Enns, "Large datasets at a glance: combining textures and colors in scientific visualization," *IEEE Trans. Vis. Comput. Graphics*, vol. 5, no. 2, pp. 145–167, 1999.

[18] P. M. Fitts, "The information capacity of the human motor system in controlling the amplitude of movement," *J. Exp. Psych.*, vol. 47, no. 6, pp. 381–391, 1954.

[19] Y. Guiard, F. Bourgeois, D. Mottet, and M. Beaudouin-Lafon, "Beyond the 10-bit barrier: Fitts' law in multi-scale electronic worlds," in *People and Computers XV—Interaction without Frontiers*. Springer, 2001, pp. 573–587.

[20] W. E. Hick, "On the rate of gain of information," *Quarterly J. Exp. Psych.*, vol. 4, no. 1, pp. 11–26, 1952.

[21] A. Cockburn, C. Gutwin, and S. Greenberg, "A predictive model of menu performance," in *Proc. SIGCHI Conf. Human Factors Comp. Sys.* ACM, 2007, p. 627–636.

[22] R. Budiu, "Scaling user interfaces: An information-processing approach to multi-device design," https://www.nngroup.com/articles/scaling-user-interfaces/, 2014, accessed on 2021-03-29.

[23] B. Brown, S. Bødker, and K. Höök, "Does HCI scale? Scale hacking and the relevance of HCI," *Interactions*, vol. 24, no. 5, p. 28–33, 2017.

[24] J. R. Ruthruff, S. G. Elbaum, and G. Rothermel, "Experimental program analysis: a new program analysis paradigm," in *Proc. Int. Symp. Software Test. Analysis*. ACM, 2006, pp. 49–60.

[25] A. Perrot, R. Bourqui, N. Hanusse, F. Lalanne, and D. Auber, "Large interactive visualization of density functions on big data infrastructure," in *Proc. IEEE Symp. Large Data Analysis Vis.* IEEE, 2015, pp. 99–106.

[26] H. Chen, R. Sukthankar, G. Wallace, and K. Li, "Scalable alignment of large-format multi-projector displays using camera homography trees," in *Proc. IEEE Vis. Conf.* IEEE, 2002, pp. 339–346.

[27] M. Rønne Jakobsen and K. Hornbæk, "Sizing up visualizations: Effects of display size in focus+context, overview+detail, and zooming interfaces," in *Proc. SIGCHI Conf. Human Factors Comp. Sys.* ACM, 2011, p. 1451–1460.

[28] M. Khoury, Y. Hu, S. Krishnan, and C. E. Scheidegger, "Drawing large graphs by low-rank stress majorization," *Comput. Graph. Forum*, vol. 31, no. 3, pp. 975–984, 2012.

[29] M. Falk and D. Weiskopf, "Output-sensitive 3D line integral convolution," *IEEE Trans. Vis. Comput. Graphics*, vol. 14, no. 4, pp. 820–834, 2008.

[30] J. L. Gustafson, "Reevaluating Amdahl's law," *Commun. ACM*, vol. 31, no. 5, pp. 532–533, 1988.

[31] M. Howison, E. W. Bethel, and H. Childs, "Hybrid parallelism for volume rendering on large-, multi-, and many-core systems," *IEEE Trans. Vis. Comput. Graphics*, vol. 18, no. 1, pp. 17–29, 2012.

[32] Y. Wang, Y. Wang, H. Zhang, Y. Sun, C.-W. Fu, M. Sedlmair, B. Chen, and O. Deussen, "Structure-aware fisheye views for efficient large graph exploration," *IEEE Trans. Vis. Comput. Graphics*, vol. 25, no. 1, pp. 566–575, 2019.

[33] M. Ghoniem, J.-D. Fekete, and P. Castagliola, "A comparison of the readability of graphs using node-link and matrix-based representations," in *Proc. IEEE InfoVis Conf.* IEEE, 2004, pp. 17–24.

[34] Y. Glémarec, A.-G. Bosser, C. Buche, J.-L. Lugrin, M. Landeck, M. E. Latoschik, and M. Chollet, "A scalability benchmark for a virtual audience perception model in virtual reality," in *Proc. ACM Symp. Virtual Reality Software Tech.* ACM, 2019, p. 48:1.

[35] J. Deng, O. Russakovsky, J. Krause, M. S. Bernstein, A. Berg, and L. Fei-Fei, "Scalable multi-label annotation," in *Proc. SIGCHI Conf. Human Factors Comp. Sys.* ACM, 2014, p. 3099–3102.

[36] L. D. Lins, J. T. Klosowski, and C. E. Scheidegger, "Nanocubes for real-time exploration of spatiotemporal datasets," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 12, pp. 2456–2465, 2013.

[37] D. Moritz, B. Howe, and J. Heer, "Falcon: Balancing interactive latency and resolution sensitivity for scalable linked visualizations," in *Proc. SIGCHI Conf. Human Factors Comp. Sys.* ACM, 2019, p. 694.

[38] J. Abello, F. van Ham, and N. Krishnan, "ASK-GraphView: A large scale graph visualization system," *IEEE Trans. Vis. Comput. Graphics*, vol. 12, no. 5, pp. 669–676, 2006.

[39] C. A. de Lara Pahins, S. A. Stephens, C. Scheidegger, and J. L. D. Comba, "Hashedcubes: Simple, low memory, real-time visual exploration of big data," *IEEE Trans. Vis. Comput. Graphics*, vol. 23, no. 1, pp. 671–680, 2017.

[40] K. Charmaz, *Constructing Grounded Theory*. SAGE, 2014.

[41] T. Isenberg, P. Isenberg, J. Chen, M. Sedlmair, and T. Möller, "A systematic review on the practice of evaluating visualization," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 12, pp. 2818–2827, 2013.

[42] D. Sacha, L. Zhang, M. Sedlmair, J. A. Lee, J. Peltonen, D. Weiskopf, S. C. North, and D. A. Keim, "Visual interaction with dimensionality reduction: A structured literature analysis," *IEEE Trans. Vis. Comput. Graphics*, vol. 23, no. 1, pp. 241–250, 2017.

[43] M. Sedlmair, C. Heinzl, S. Bruckner, H. Piringer, and T. Moller, "Visual parameter space analysis: A conceptual framework," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 12, pp. 2161–2170, 2014.

[44] P. Isenberg, F. Heimerl, S. Koch, T. Isenberg, P. Xu, C. Stolper, M. Sedlmair, J. Chen, T. Möller, and J. Stasko, "vispubdata.org: A metadata collection about IEEE Visualization (VIS) publications," *IEEE Trans. Vis. Comput. Graphics*, vol. 23, no. 9, pp. 2199–2206, 2017.

[45] E. M. Bennett, R. Alpert, and A. C. Goldstein, "Communications through limited-response questioning," *Publ. Opinion Quarterly*, vol. 18, no. 3, pp. 303–308, 1954.

[46] L. McInnes and J. Healy, "UMAP: Uniform manifold approximation and projection for dimension reduction," *CoRR*, vol. abs/1802.03426, 2018.

[47] T. Munzner, "A nested process model for visualization design and validation," *IEEE Trans. Vis. Comput. Graphics*, vol. 15, no. 6, pp. 921–928, 2009.

[48] J. Jo, F. Vernier, P. Dragicevic, and J. Fekete, "A declarative rendering model for multiclass density maps," *IEEE Trans. Vis. Comput. Graphics*, vol. 25, no. 1, pp. 470–480, 2019.

[49] A. Lex, N. Gehlenborg, H. Strobelt, R. Vuillemot, and H. Pfister, "UpSet: Visualization of intersecting sets," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 12, pp. 1983–1992, 12 2014.

[50] J. Tierny, A. Gyulassy, E. Simon, and V. Pascucci, "Loop surgery for volumetric meshes: Reeb graphs reduced to contour trees," *IEEE Trans. Vis. Comput. Graphics*, vol. 15, no. 6, pp. 1177–1184, 2009.

[51] M. Hadwiger, J. Beyer, W. Jeong, and H. Pfister, "Interactive volume exploration of petascale microscopy data streams using a visualization-driven virtual memory approach," *IEEE Trans. Vis. Comput. Graphics*, vol. 18, no. 12, pp. 2285–2294, 2012.

[52] J. Beyer, A. K. Al-Awami, N. Kasthuri, J. W. Lichtman, H. Pfister, and M. Hadwiger, "ConnectomeExplorer: Query-guided visual analysis of large volumetric neuroscience data," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 12, pp. 2868–2877, 2013.

[53] C. Y. Ip and A. Varshney, "Saliency-assisted navigation of very large landscape images," *IEEE Trans. Vis. Comput. Graphics*, vol. 17, no. 12, pp. 1737–1746, 2011.

[54] K. Kurzhals, M. Hlawatsch, F. Heimerl, M. Burch, T. Ertl, and D. Weiskopf, "Gaze Stripes: Image-based visualization of eye tracking data," *IEEE Trans. Vis. Comput. Graphics*, vol. 22, no. 1, pp. 1005–1014, 2016.

[55] W. Dou, X. Wang, R. Chang, and W. Ribarsky, "ParallelTopics: A probabilistic approach to exploring document collections," in *Proc. IEEE Conf. Visual Analytics Sci. Tech.* IEEE, 2011, pp. 231–240.

[56] C. Papadopoulos, I. Gutenko, and A. E. Kaufman, "VEEVVIE: visual explorer for empirical visualization, VR and interaction experiments," *IEEE Trans. Vis. Comput. Graphics*, vol. 22, no. 1, pp. 111–120, 2016.

[57] J. Georgii and R. Westermann, "A generic and scalable pipeline for GPU tetrahedral grid rendering," *IEEE Trans. Vis. Comput. Graphics*, vol. 12, no. 5, pp. 1345–1352, 2006.

[58] W. Qiao, M. McLennan, R. Kennell, D. S. Ebert, and G. Klimeck, "Hub-based simulation and graphics hardware accelerated visualization for nanotechnology applications," *IEEE Trans. Vis. Comput. Graphics*, vol. 12, no. 5, pp. 1061–1068, 2006.

[59] H. Chen, R. Sukthankar, G. Wallace, and K. Li, "Scalable alignment of large-format multi-projector displays using camera homography trees," in *Proc. IEEE Vis. Conf.* IEEE, 2002, pp. 339–346.

[60] E. T. Brown, J. Liu, C. E. Brodley, and R. Chang, "Dis-function: Learning distance functions interactively," in *Proc. IEEE Conf. Visual Analytics Sci. Tech.* IEEE, 2012, pp. 83–92.

[61] J. Jo, J. Huh, J. Park, B. H. Kim, and J. Seo, "LiveGantt: interactively visualizing a large manufacturing schedule," *IEEE Trans. Vis. Comput. Graphics*, vol. 20, no. 12, pp. 2329–2338, 2014.

[62] J. Heer and M. Bostock, "Declarative language design for interactive visualization," *IEEE Trans. Vis. Comput. Graphics*, vol. 16, no. 6, pp. 1149–1156, 2010.

[63] V. Yoghourdjian, Y. Yang, T. Dwyer, L. Lawrence, M. Wybrow, and K. Marriott, "Scalability of network visualisation from a cognitive load perspective," *IEEE Trans. Vis. Comput. Graphics*, vol. 27, no. 2, pp. 1677–1687, 02 2021.

[64] H. Strobelt, B. Alsallakh, J. Botros, B. Peterson, M. Borowsky, H. Pfister, and A. Lex, "Vials: visualizing alternative splicing of genes," *IEEE Trans. Vis. Comput. Graphics*, vol. 22, no. 1, pp. 399–408, 2016.

[65] N. Pezzotti, J. Thijssen, A. Mordvintsev, T. Höllt, B. van Lew, B. P. F. Lelieveldt, E. Eisemann, and A. Vilanova, "GPGPU linear complexity t-SNE optimization," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 1, pp. 1172–1181, 2020.

[66] R. Veras and C. Collins, "Discriminability tests for visualization effectiveness and scalability," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 1, pp. 749–758, 2020.

[67] W. Aigner, S. Hoffmann, and A. Rind, "EvalBench: A software library for visualization evaluation," *Comput. Graph. Forum*, vol. 32, no. 3, pp. 41–50, 2013.

[68] A. Eiselmayer, C. Wacharamanotham, M. Beaudouin-Lafon, and W. E. Mackay, "Touchstone2: An interactive environment for exploring trade-offs in HCI experiment design," in *Proc. SIGCHI Conf. Human Factors Comp. Sys.* ACM, 2019, p. 1–11.

[69] C. Schulz, A. Nocaj, M. El-Assady, S. Frey, M. Hlawatsch, M. Hund, G. K. Karch, R. Netzel, C. Schätzle, M. Butt, D. A. Keim, T. Ertl, U. Brandes, and D. Weiskopf, "Generative data models for validation and evaluation of visualization techniques," in *Proc. Workshop Beyond Time Errors Novel Eval. Meth. Vis.* ACM, 2016, pp. 112–124.

[70] F. Schreiber and D. Weiskopf, "Quantitative visual computing," *it – Informat. Tech.*, vol. 64, no. 4–5, pp. 119–120, 2022.

[71] J.-D. Fekete and J. Freire, "Exploring reproducibility in visualization," *IEEE Comput. Graph. Appl.*, vol. 40, no. 5, pp. 108–119, 2020.

[72] C. Plaisant, J.-D. Fekete, and G. Grinstein, "Promoting insight-based evaluation of visualizations: From contest to benchmark repository," *IEEE Trans. Vis. Comput. Graphics*, vol. 14, no. 1, pp. 120–134, 2008.

[73] L. Battle, P. Eichmann, M. Angelini, T. Catarci, G. Santucci, Y. Zheng, C. Binnig, J.-D. Fekete, and D. Moritz, "Database benchmarking for supporting real-time interactive querying of large data," in *Proc. ACM SIGMOD Int. Conf. Manag. Data*. ACM, 2020, pp. 1571–1587.

[74] K.-L. Ma, "In situ visualization at extreme scale: Challenges and opportunities," *IEEE Comput. Graph. Appl.*, vol. 29, no. 6, pp. 14–19, 2009.

[75] J.-D. Fekete, D. Fisher, A. Nandi, and M. Sedlmair, "Progressive Data Analysis and Visualization (Dagstuhl Seminar 18411)," *Dagstuhl Reports*, vol. 8, no. 10, pp. 1–40, 2019.

[76] L. Wilkinson and G. Wills, "Scagnostics distributions," *J. Comput. Graph. Stat.*, vol. 17, no. 2, pp. 473–491, 2008.

[77] J. Slack, K. Hildebrand, and T. Munzner, "PRISAD: a partitioned rendering infrastructure for scalable accordion drawing," in *Proc. IEEE InfoVis Conf.*, 2005, pp. 41–48.

Gaëlle Richer is a postdoctoral researcher at Inria Saclay, Aviz research team. She received her PhD in Computer Science in 2019 from the University of Bordeaux, France where she worked at the LaBRI. Her research interests include Interactive Visualization and Visual Analytics with a focus on scalability issues.

Alexis Pister is a second year Ph.D. student at Inria Saclay Aviz team and Telecom Paris. His subject consists in developing visual analytics tools for the use of historians and sociologists for Social Network Analysis with an emphasis on interpretability and user experience. His research interests are Visual Analytics and Network Analysis.

Photo: Max Kovalenko

Moataz Abdelaal is a fourth year Ph.D. student at the Visualization Research Center (VISUS) at University of Stuttgart. His research interests are Network Visualization, Visual Analytics, and Human-Computer Interaction.

Jean-Daniel Fekete is the Scientific Leader of the INRIA Project Team Aviz that he founded in 2007. He received his PhD in Computer Science in 1996 from University of Paris-Sud, France. His main research areas are Visual Analytics, Information Visualization and Human Computer Interaction. He is a Senior Member of IEEE.

Michael Sedlmair is a professor at the University of Stuttgart and heads the VisVAR research group there. He received his Ph.D. degree in Computer Science in 2010 from the Ludwig Maximilians University of Munich, Germany. His research interests focus on Visualization/Visual Analytics, Human-Computer Interaction, and Augmented/Virtual Reality.

Photo: Max Kovalenko

Daniel Weiskopf is a Professor at the Visualization Research Center (VISUS) of the University of Stuttgart, Germany. He received his Dr. rer. nat. degree (similar to PhD) in Physics from the University of Tübingen, Germany, in 2001, and the Habilitation degree in Computer Science from the University of Stuttgart in 2005. His research interests include Visualization, Eye Tracking, Human-Computer Interaction, Computer Graphics, and Special and General Relativity.