# StressDiffVis: Visual Analytics for Multi-Model Stress Comparison

Jiabao Huang, Zikun Deng, Hanlin Song, Xiang Chen, Shaowu Gao, and Yi Cai

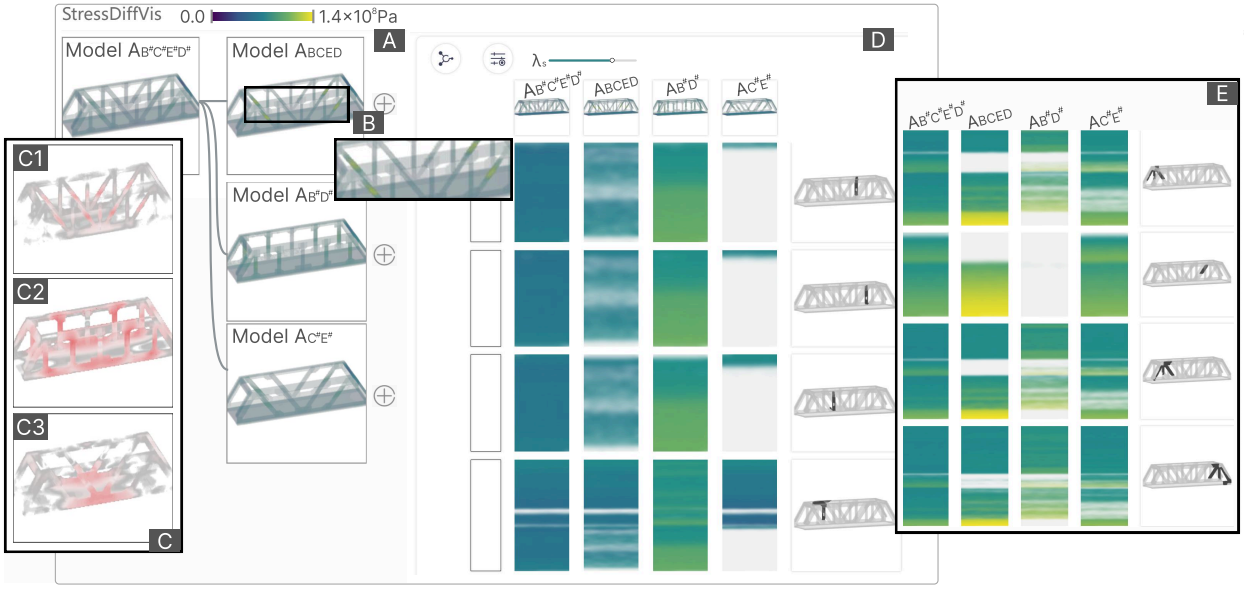


Fig. 1: Case study of a truss bridge. (A) The tree view shows four alternatives with the left one as the base model. (B) Yellow region indicating significant stress concentration in the diagonal supports of Model $A_{BCED}$. (C) The difference-aware visualization of (C1) Models $A_{BCED}$, (C2) $A_{B^{\#}D^{\#}}$, and (C3) $A_{C^{\#}E^{\#}}$, respectively, compared to the base model. (D) The comparison view presents the stress data of the top four segments (corresponding to vertical supports) across alternatives, where green indicates stress magnitudes within an acceptable range. (E) Four segments corresponding to the diagonal support of (B) exhibit high stress, indicated by the yellow.

**Abstract**—Structural analysis is essential in modern industrial design, where engineers iteratively refine geometry models based on stress simulations to achieve optimized designs. However, comparing stress distributions across multiple model variants remains challenging due to the complexity of stress fields, which are high-dimensional, unevenly distributed, and dependent on intricate geometric structures. Existing tools primarily support single-model analysis and lack dedicated functionalities for multi-model comparison. As a result, engineers must rely on manual, cognitively demanding visual inspections, making it difficult to systematically identify and interpret stress variations across design iterations. To address these limitations, we propose StressDiffVis, a visual analytics approach that facilitates stress field comparison across multiple structural models. StressDiffVis employs a volumetric representation to encode stress distributions while minimizing occlusion, enabling voxel-wise difference analysis for model comparison. To support localized analysis, we introduce model segmentation, grouping voxels with similar stress patterns across models. StressDiffVis integrates these techniques into an interactive interface with a tree view, organizing models by the iterative design process, and a comparison view, using a matrix layout for detailed comparisons. We demonstrate the effectiveness of StressDiffVis through two case studies illustrating its utility in comparative stress analysis. In addition, expert interviews confirm its potential to enhance engineering workflows.

**Index Terms**—Stress analysis, comparative visualization, volume visualization

## 1 Introduction

In modern industrial domains such as manufacturing and construction, computer-aided structural design and stress simulation play a pivotal role. Structural engineers begin by creating an initial model and iteratively refining it based on simulation results to address deficiencies, ultimately achieving an optimized design [5, 52]. For example, if a simulation under a specific load condition reveals a risk of fracture at the midspan of a bridge, additional support should be introduced at the midspan, or a more robust material should be used to enhance structural stability. In recent years, automated optimization algorithms have been increasingly employed to generate alternative solutions based on identified structural deficiencies, enabling engineers to explore a wider range of design possibilities [32, 56]. However, engineers still strongly demand informed decision-making support for selecting the most suitable model variant through stress analysis comparison [1, 40].

Despite its importance, this task remains under-supported. Existing stress analysis software, such as ANSYS and ABAQUS, lacks the functionality for comparing multiple stress fields across different model variants, thus requiring users to rely on side-by-side visual inspection. Akin to that in the "Find the Difference" game, engineers must manually identify subtle variations in stress fields across similar models, making the comparison process cognitively demanding and error-prone. While various comparative visualization techniques for 3D models have been proposed in academia, they primarily focus on parameter

- *Jiabao Huang, Zikun Deng, Hanlin Song, Xiang Chen, and Yi Cai are with the School of Software Engineering, South China University of Technology, and Key Laboratory of Big Data and Intelligent Robot (South China University of Technology), Ministry of Education. E-mail: {sejiabaoh, sehanlinsong, secx2023software}@mail.scut.edu.cn, {zkdeng, ycai}@scut.edu.cn. Zikun Deng is the corresponding author.*
- *Shaowu Gao is with the Greater Bay Area National Center of Technology Innovation. E-mail: gaoshaowu@ncti-gba.cn.*

*Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx*

comparison [29], model meshes [10], or structural geometry [14], without integrating stress field analysis. Academic efforts have proposed glyph-based approaches [30, 36] to visualize differences in stress fields. However, these approaches suffer from high interaction overhead and visual clutter due to irregular glyph placement. Other techniques simplify stress fields by extracting curves or slices [6, 28]. However, these abstractions sacrifice the structural context, which limits interpretability in stress analysis.

Motivated by the observed limitations, we propose a visual analytics approach that enables engineers to efficiently compare multiple structural models based on stress analysis. Developing such an approach is non-trivial due to the following challenges.

First, stress is unevenly distributed across a model, requiring visualization techniques that convey both global stress patterns and localized stress concentrations. The complex geometry of structural models exacerbates occlusion and navigation challenges in a 3D environment, further complicating effective visualization design. Thus, a less obstructive visualization method coupled with a model navigation method is needed to support both global and local stress analysis of a single model. Second, when multiple models are involved, the comparative analysis must integrate both global and local stress distributions to facilitate meaningful comparisons. Manually inspecting models one by one while mentally tracking stress variations across different scales is cognitively demanding. To mitigate this challenge, a unified representation that explicitly encodes inter-model stress differences, combined with a well-structured visualization, is essential to avoid turning the comparison process into a "*Find the Differences*" task.

In this study, we propose StressDiffVis to address the aforementioned challenges. First of all, we transform the stress field of the model into a unified **volumetric representation**. This way enables the visualization of each model's stress distribution using occlusion-minimizing volume rendering [8]. Volumetric representation can also facilitate the comparison of multiple models as it allows for the derivation of voxel-wise differences and, thereby, difference-aware volume visualizations can be applied, again showing its advantage. Based on the volumetric representation, we can further apply **model segmentation** based on stress behaviors across models for localized stress analysis and comparison. In each segment, voxels exhibit similar stress patterns or variations across different models. By visually analyzing segments, both intra-model variations and inter-model differences can be effectively identified.

By coupling the volume-based solution and model segmentation, we develop a visual analytics approach named StressDiffVis. It mainly comprises two views, namely, the tree view and the comparison view. In the tree view, all models to be compared are well-organized in a tree structure following the iterative design process and are rendered with volume visualization in either stress-aware or difference-aware mode. In the comparison view, the model segments versus the model variants are visualized in a matrix-based layout with each cell encoding the segment stress patterns of a model variant. The row-by-row and column-by-column juxtaposition supports both inter-model and intra-model localized comparison. We present two case studies, one for the iterative design of a wooden chair and the other for a truss bridge, showing the effectiveness of StressDiffVis. We also perform expert interviews, collecting positive feedback and further suggestions to improve StressDiffVis. In sum, our contributions are as follows:

- We couple a volume-based solution and model segmentation for comparing multiple models globally and locally on stress fields.
- We develop a visual analytics approach named StressDiffVis that instantiates these approaches for engineers to efficiently compare multiple structural models based on stress analysis.
- We evaluate StressDiffVis with two case studies and expert interviews, demonstrating StressDiffVis's effectiveness.

## 2 Related Work

We review prior studies on structure design, stress visualization, visual comparison in structural design, and visual comparison of 3D fields.

### 2.1 Structure Design

Iteration is a common case in traditional structure design processes, wherein designers typically refine and evaluate multiple alternatives through discussion and testing before finalizing a structure [5, 16, 26, 52]. This iterative process ensures that the final design meets structural integrity, aesthetic, and functional requirements. To facilitate this process, automated techniques have been developed to optimize structures, such as by reducing stress concentrations and minimizing material costs [35, 41, 45, 55]. Recent advancements in generative design leverage computational algorithms to explore diverse design variations [5, 40, 48, 52]. Ultimately, designers must evaluate multiple alternatives to identify the most suitable one.

In this study, we explore visualization techniques to evaluate multiple alternatives, with a particular focus on geometric structure designs relying on stress analysis.

### 2.2 Stress Visualization

Stress visualization plays a crucial role in structural analysis, enabling engineers to interpret the simulation results of structural modifications with the aid of finite element analysis. Contour plots are widely used to visualize scalar stress fields and are standard in tools like Abaqus and ANSYS. They map stress magnitudes to surface colors with color gradients, offering an intuitive overview of stress distributions. Direct volume rendering (DVR) enables the visualization of internal stress variations within a volumetric domain [12, 17, 25], providing more comprehensive insights. To preserve directional information of stress, glyph-based visualization [24, 30, 36, 38] and line-based visualization [4, 11, 26, 34, 43] are developed. Glyphs are 3D geometric icons that encode tensor attributes through their shape, size, color, and orientation. Line-based approaches use streamlines whose tangent directions represent principal stress directions, while thickness and color encode stress magnitude.

This study focuses on mechanical structure design, where the analyst needs to evaluate the overall distribution of stress magnitudes, with particular emphasis on high-stress regions and stress concentrations. We adopt von Mises stress [31] as a scalar measure to assess stress magnitudes. Von Mises stress abstracts away directional information and assumes isotropic, ductile material behavior. While it does not account for anisotropy or tension–compression asymmetry, this assumption remains appropriate for most structural analysis scenarios [3, 35, 41, 55]. We employ volume visualization and leverage the discrete, regular, and computable nature of volume representations to enable effective multi-alternative comparative analysis.

### 2.3 Visual Comparison in Structural Design

Visual comparison is essential for informed decision-making [7, 27, 44]. During structural design, the geometric structure, stress distribution, and quantitative metrics evolve and require comparison.

**Geometric structure.** During manual design and automated optimization, model structures continuously evolve. Various approaches have been developed for geometric structural comparison [2, 10, 14, 37], typically merging models and employing color encoding to highlight differences. For example, 3D Diff [13] categorizes model differences into four types—unmodified, modified, added, and conflicting regions—and applies distinct colors for differentiation.

**Stress distribution.** Comparative visualization of stress distributions in geometric models remains underexplored [22]. Existing stress analysis software (e.g., ANSYS and ABAQUS) lacks native support for comparative stress distribution analysis across multiple alternatives, often relying on juxtaposed pairwise comparisons. Related techniques in biomedical visualization provide valuable insights, with glyph-based methods being the most common [30, 42, 53]. These approaches encode local differences by placing glyphs on the model, but they introduce several challenges: (1) cognitive load for glyph interpretation, (2) frequent perspective adjustments (e.g., zooming and rotation), and (3) irregular glyph placement, which hampers effective stress pattern analysis. Beyond glyph-based techniques, Dick et al. [11] proposed a line-based encoding to visualize normal and shear stress differences in femurs before and after implant surgery. However, it is limited to pairwise comparisons, restricting its applicability in multi-alternative scenarios.

**Quantitative metric.** For handling numerous alternatives, defining quantitative metrics facilitates exploration, filtering, and comparison. Dream Lens [29] exemplifies this approach. However, such methods

risk information loss by abstracting away the 3D geometry and spatial context of stress distributions.

Our study focuses on comparing stress distributions in the context of structural design. In the following part, we discuss how this comparison can be supported from a visualization perspective.

### 2.4 Visual Comparison of 3D Fields

In this study, we focus on structural analysis and design, where comparative analysis of stress distributions is crucial. The stress distribution of a 3D structure can be viewed as a 3D field. Gleicher [19] summarized visual designs of comparison into three categories, namely, juxtaposition, superposition, and explicit encoding. Below, we narrow down our scope to discuss how visual comparison design and strategies can be applied to the comparison of 3D fields.

**Juxtaposition**. The category means that the targets to be compared are placed side by side, and the user compares them by scanning them. Such a manner can be commonly seen in existing stress analysis software (e.g., ANSYS and ABAQUS). In this way, users need to maintain different objects in their minds, which makes it difficult to expand to multiple objects. The occlusion nature in 3D environments further makes the comparison ineffective. Some studies have proposed effective means that ease the issue of occlusion and improve scalability. For example, slices from 3D scalar fields can be extracted as representatives for comparison [28]. Furthermore, 1D lines can be extracted from 3D scalar fields using space-filling curves, such as the Hilbert curve, and aligned within a 2D layout for side-by-side comparison [6, 49]. While these methods effectively preserve voxel-level spatial adjacency during the transformation from 3D to 2D, they fail to maintain the structural context necessary for understanding semantic regions within the model.

**Superposition and Explicit encoding.** The direct superposition of multiple fields often causes visual confusion, making it difficult to distinguish the individual fields. To solve this problem, researchers often use glyphs to summarize the difference between the data from different fields [30, 42, 53]. Such a manner can be considered a combination of superposition and explicit encoding. For example, Zhang et al. [53] designed a glyph that exploits the shape, size, and orientation channels, and the glyphs from different fields can be superimposed to show differences between fields. As noted earlier, glyph-based methods involve high cognitive load, frequent view changes, and irregular placement, making stress analysis difficult.

In this study, we explore an alternative explicit encoding approach for comparing stress distributions. After voxelizing the stress fields, we compute pairwise differences and encode them directly to reveal spatial variations. To enable effective comparison across multiple design alternatives, inspired by the *Select Subset* and *Scan Sequentially* strategies [19], we further propose a model segmentation technique and a matrix-based view that juxtaposes alternatives and segments. This approach supports both localized and global stress analysis at both intra-model and inter-model levels.

## 3 Background

This section introduces the concepts used in this paper, the collaboration with domain experts, and finally, the requirement analysis.

### 3.1 Concepts

We formally define the key concepts used throughout the paper.

- **Nodal stress data.** Nodal stress data refers to the set of stress values computed at discrete nodes within a finite element model. These data capture the localized stress distribution across the structure and serve as the basis for evaluating mechanical performance, identifying critical stress concentrations, and guiding design modifications.
- **Base model.** The base model is the initial, unmodified three-dimensional geometric representation of a structure, serving as the reference configuration for subsequent modifications. It defines the spatial and structural characteristics of the design and undergoes finite element analysis (FEA) to generate baseline nodal stress data, which informs the necessity and direction of structural modification.
- **Modification.** A modification is a targeted design alteration applied to the base model with the aim of improving structural performance. Modifications may involve geometric adjustments, material changes, or reinforcement additions, each of which affects the stress distribution and mechanical response of the model.
- **Design alternative.** A design alternative denotes a specific modified variant of the base model, incorporating one or more design modifications. It is also essentially a geometric representation. Note that the base model is also one of the alternatives. Since all alternatives originate from the same base model and are derived through incremental modifications, they exhibit an inherent hierarchical relationship. That is, an alternative may inherit structural characteristics from its predecessor while introducing new design modifications. Each alternative is independently subjected to finite element analysis (FEA) to obtain updated nodal stress data, enabling comparative evaluation across different design iterations.

### 3.2 Collaboration with Domain Experts

Over the past six months, we have collaborated with three domain experts with extensive experience in structural analysis and design. Expert 1 (E1) has 20 years of experience in structural analysis and leads a team of 20 researchers or engineers developing computer-aided engineering (CAE) software for stress-based analysis. Expert 2 (E2) and Expert 3 (E3) have 2 and 5 years of experience in civil engineering, respectively, with substantial expertise in structural design. During the collaboration, we held meetings every two weeks with the three experts to discuss stress-based structural design. The discussions covered various aspects, including but not limited to design workflows, application scenarios, software tools, and visualization approaches.

In the early stages of the collaboration, we identified a key challenge in structural design: iterative design is usually required, but there is a lack of an effective means for comparing multiple designs (alternatives). In the later stages, our discussions focused on this challenge. After reviewing existing comparative visualization and stress visualization techniques, we opted to design a visual analytics approach to address the challenge. We asked E3 to provide a simple model (a wooden chair) and its variants so that we could design and develop a visual analysis approach. We iteratively proposed solutions—including algorithms, visualizations, and interactions—through sketches and system prototypes, continuously gathering expert feedback to refine the requirements.

### 3.3 Requirement Analysis

The requirements can be summarized into two aspects, namely, stress analysis and comparative analysis.

**Stress Analysis.** Stress analysis is the most fundamental part of the entire design process. As the 3D model can be geometrically large and structurally complex, stress analysis should be performed from overall and localized perspectives.

- **R1: Overall stress analysis.** An overall perspective on stress distribution is critical for assessing the structural integrity of a model. Engineers and designers need an intuitive visualization of stress distribution to identify load-bearing structures, evaluate overall material efficiency, and detect potential failure zones. Both high- and low-stress regions provide critical insights, guiding decisions on material improvement and structural reinforcement.
- **R2: Localized stress analysis.** While an overall view provides a broad understanding, detailed analysis of high-stress areas is equally important. Stress concentrations often occur near joints, sharp edges, or load-bearing points, where material failure is most likely. A localized visual analysis helps designers focus on these critical areas, enabling precise modifications that enhance structural durability and safety. Without such an analysis, small yet significant design flaws may persist, compromising performance.

**Comparative Analysis.** Effective comparative analysis is a critical step toward informed decision-making. Given the variations in stress distribution within a single alternative and between different alternatives, the comparative analysis can be further refined into the following two requirements.

- **R3: Intra-Model comparison.** Within a single alternative model, stress distribution can vary significantly across different regions due to geometry, material properties, and applied loads. Analyzing these variations through visual comparison is crucial for

identifying potential weak points, ensuring balanced load distribution, and optimizing material usage. For example, areas with excessive stress may require reinforcement, while regions experiencing minimal stress could be redesigned to reduce material consumption. By providing a visual representation of stress differences within the model, engineers can make more informed design modifications to enhance structural efficiency and integrity.

- **R4: Inter-Model comparison.** When multiple alternative models are considered, a comparison across alternatives is necessary to determine the most viable one. Different alternatives may exhibit varying stress patterns and structural efficiencies, making it essential to visualize these differences clearly. For example, one design may reduce peak stress concentrations but require more material, while another may achieve a more uniform stress distribution with lower weight. By systematically comparing these factors, engineers can identify the optimal balance between performance, cost, and integrity.

## 4 System

This section introduces StressDiffVis, a novel system for effective comparative stress analysis of multiple designed models.

### 4.1 System Overview

StressDiffVis mainly serves as an additional interface to facilitate users in performing comparative stress analyses of models during the model design process (Figure 2A).

The visual interface of StressDiffVis mainly comprises two coordinated views: the tree view (Figure 2C) and the comparison view (Figure 2E). In the tree view, the user can import alternatives, and these alternatives are organized as a tree-structured layout, following the design inheritance relationship between alternatives. Each alternative with stress distribution is transformed into a volumetric representation (Figure 2B). By default, a stress-aware volume visualization is used to present the stress distribution within alternatives. The user can turn to difference-aware volume visualization for analyzing the stress differences among alternatives. The comparative view facilitates a finer-grained comparison of multiple alternatives. The model is segmented into multiple parts based on their stress data across all alternatives so that each model segment will exhibit similar stress performance (Figure 2D). Afterwards, the stress data of segments across alternatives are visualized in a matrix-like visualization, supporting segment-to-segment and alternative-to-alternative comparison.

The three computational modules, namely, volumetric transformation, differential computation, and model segmentation, are running in a backend server, supporting seamless interactions in StressDiffVis.

### 4.2 Transformation

First of all, for each alternative imported into the StressDiffVis, the nodal stress data associated with the geometric representation should be transformed to a volumetric representation. The data transformation pipeline of StressDiffVis is illustrated in Figure 2B. For simplicity, the geometric representation is depicted in 2D as a triangle in Figure 2B, although the actual process operates in 3D.

Nodal stress data for each alternative are obtained via finite element analysis (FEA), which generates a set of nodes within the geometric representation and estimates the stress value of every node, as depicted in the upper triangle of Figure 2B. To convert this data into a structured volumetric representation suitable for analysis, we apply tricubic interpolation directly to the nodal stress data, generating a smooth and continuous stress distribution over a voxel grid, as shown in the lower triangle of Figure 2B. Once the interpolation is completed, we get volumetric stress data, the primary data foundation of StressDiffVis.

The reasons for using the volume-based representation are as follows. First, compared to contour plots, the volume visualization can reveal critical internal stress regions (such as stress concentrations inside thick components) and reduce the need for users to frequently navigate and zoom to explore high-stress regions. Second, the volumetric representation provides a common spatial framework, making the stress difference across different alternatives to be calculated.

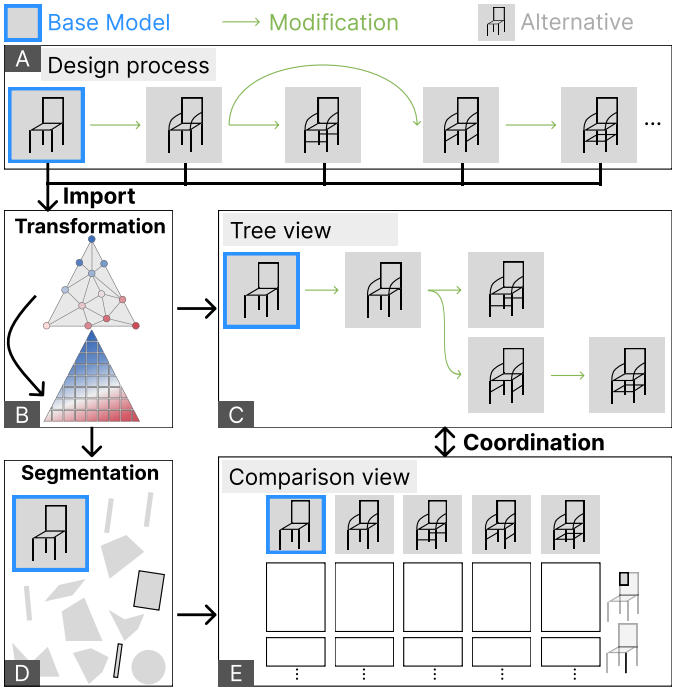


Fig. 2: Pipeline of StressDiffVis illustrated with a chair design. (A) A chair with a blue border (the base model) is being modified, generating multiple alternatives. (B) For each imported model, the geometric representation with nodal stress data (top) is transformed into a volumetric representation with interpolated stress value (bottom). (C) The tree view renders the volumetric representation using either stress-aware or difference-aware volume visualization. (D) The segmentation process divides the geometric structure into multiple segments. (E) The comparison view supports segment-by-segment and alternative-by-alternative comparison.

### 4.3 Tree View

The tree view presents multiple alternatives in a hierarchical tree-structured layout, following the inheritance relationship of the alternatives. It provides two rendering modes for alternatives, namely, stress-aware and difference-aware volume visualization. Below, we introduce the tree structure layout, stress-aware volume visualization, and difference-aware volume visualization.

#### 4.3.1 Hierarchical Tree-structured Layout

To support iterative design, it is important to represent the evolution of design alternatives. This goal aligns with analytical provenance visualization [9, 33, 50], which has explored various ways to capture and organize iterative processes.

Following prior studies [21, 39], we adopt a tree-based layout to represent the design process, where each node denotes an alternative, and edges capture inheritance relationships between them. To visualize this tree structure, we adopt a left-to-right hierarchical layout combined with a grid-based arrangement. Each "child" alternative is positioned to the right of its "parent" alternative and is connected via a straight line or Bézier curve to illustrate the hierarchical relationship. Each node supports two rendering modes to visualize the corresponding alternative's stress, which will be introduced later.

Users can add or remove design alternatives. The first alternative, corresponding to the base model, is placed at the top-left corner by default. By clicking the button located below or to the right of an existing alternative, users can append a new alternative to its bottom or right, respectively, thereby explicitly defining the inheritance relationship. This approach facilitates the flexible management of the design process.

#### 4.3.2 Stress-aware Volume Visualization

By default, the volumetric stress data of each alternative is rendered within the node by volume visualization, which allows users to perceive stress distribution in detail within a 3D geometric model.

**Volume rendering.** We employ ray marching [20] for volume visualization. In this approach, a ray is cast from each screen pixel into the volumetric stress data that encodes the 3D stress field of the model. As the ray traverses the volume, it samples stress values from intersected voxels in a front-to-back manner. These sampled values are then accumulated to compute the final pixel value, which is subsequently mapped to a specific color with transparency using a transfer function.

**Transfer function.** In stress analysis, high-stress regions are of primary interest because they are critical for structural improvement. Thus, the transfer function is designed to enhance the visibility of the high-stress regions. Specifically, we adopt the Viridis color scheme, which is colorblind-friendly and features a blue-to-yellow gradient. We assign yellow hues with higher opacity to the voxels with the failure stress value, indicating regions with an increased likelihood of failure. Conversely, low-stress regions are mapped to blue hues with lower opacity, with zero-stress areas rendered fully transparent, leaving users to focus on critical stress concentrations.

#### 4.3.3 Difference-aware Volume Visualization

The difference rendering mode can be triggered by selecting a reference alternative in the tree view. By visualizing the difference between the reference and other alternatives, users can compare alternatives by perceiving the difference in stress distribution between them.

**Differential computation.** Owing to the unified volumetric representation, the stress difference can be computed as follows. We denote the reference alternative $\hat{a}$ and other alternatives $A = \{a_1, a_2, \ldots, a_m\}$, where $\hat{a} \notin A$. The volumetric stress data of the reference is denoted as $\hat{V}$, while the volumetric stress data of the other alternative $a_i$ is denoted as $V_i$. The differential stress between $\hat{a}$ and $a_i$ denoted as $\Delta V_i$, can be computed as $V_i \ominus \hat{V}$. Particularly, $\ominus$ means the voxel-wise subtraction operation based on voxel spatial alignment. Here, we assume that all models are pre-aligned, meaning that their geometric representations across alternatives share a consistent spatial frame with the reference. Mathematically, for each voxel located at $(x,y,z)$, the resulting voxel value is computed as:

$$\Delta V_i(x,y,z) = V_i(x,y,z) - \hat{V}(x,y,z),$$

where $V_i(x,y,z)$ and $\hat{V}(x,y,z)$ denote the voxel values at the same spatial coordinates in $V_i$ and $\hat{V}$, respectively. Notably, voxels with a value equal to zero, indicating an empty region outside the model, are excluded from the computation. In this way, we ensure that differential computation is performed on the regions shared by the models, which are assumed to be the primary focus of the structural improvement process. Finally, we obtain volumetric difference data $\{\Delta V_1, \Delta V_2, ..., \Delta V_m\}$ for difference-aware volume visualization.

**Transfer function.** In this context, regions with large stress differences are of primary interest as they indicate significant deviations from the reference alternative, either due to substantial improvement or deterioration. In contrast, regions with small differences are of lesser concern as they exhibit minor variations relative to the reference. To effectively convey these distinctions, the transfer function employs a gradient color mapping as follows. Larger positive differences are assigned redder hues with increasing opacity, highlighting regions of deterioration. Conversely, lower negative differences are assigned to deeper blue hues with higher opacity, emphasizing regions of improvement. Near-zero differences are mapped to transparent gray, thereby reducing their visual prominence and ensuring focus remains on areas with meaningful change.

### 4.4 Comparison View

The comparison view enables users to systematically analyze and compare stress differences across multiple alternatives. To facilitate a localized comparison of stress differences, the model is partitioned into multiple segments based on the stress patterns across all alternatives. Given the structure difference between alternatives, the segmentation is performed on their shared structure instead of the base model. The resulting segments, along with the voxels within them, are visualized through an alternative-segment matrix, allowing for row-by-row and column-by-column comparison of stress differences across different segments and alternatives. We detail the segmentation algorithm and the design of the alternative–segment matrix below.

#### 4.4.1 Segmentation

Following the *Select Subset* strategy proposed by Gleicher [19], model segmentation is primarily introduced to facilitate localized stress analysis. For intra-model analysis in a single alternative, we want to identify those geometry regions where voxels exhibit similar stress values. Furthermore, for inter-model comparison, we want the segmented regions to exhibit stress similarity in each alternative, respectively. In this way, a comprehensive alternative comparison can be performed by comparing the stress within individual segments one by one. For example, a part that shows high stress (indicated by high voxel values) in the first alternative but low stress in others indicates that some modification of the first alternative introduced localized structural weaknesses. Thus, we propose the following model segmentation approach.

Recall that the volumetric representations of multiple alternatives are denoted as $\{V_1, V_2, ...V_m\}$. we define a vector $\mathbf{v}$ for each voxel at position $(x,y,z)$ as $\mathbf{v} = (V_1(x,y,z), V_2(x,y,z), \ldots, V_m(x,y,z))$, where $V_i(x,y,z)$ represents the voxel stress value in $i$-th alternative $a_i$ at $(x,y,z)$. Similarly, we define a vector $\Delta\mathbf{v}$ for each voxel at position $(x,y,z)$ as $\Delta\mathbf{v} = (\Delta V_1(x,y,z), \Delta V_2(x,y,z), \ldots, \Delta V_m(x,y,z))$, where $\Delta V_i(x,y,z)$ represents the voxel stress difference in $i$-th alternative $a_i$ at $(x,y,z)$.

The first step is **discretization**, which aims to obtain a meaningful representation of every voxel.

*Stress-aware discretization.* For stress value, we discretize voxel stress values into high-stress or low-stress labels based on a given stress threshold $\lambda_s$. $\lambda_s$ is determined based on the distribution of stress values. For each voxel at location $(x,y,z)$ in the $i$-th alternative, the stress value $V_i(x,y,z)$ is descretized as follows:

$$l_i = \begin{cases} -1, & V_i(x,y,z) < \lambda_s \quad \text{/* low stress */} \\ 1, & V_i(x,y,z) \geq \lambda_s \quad \text{/* high stress */} \end{cases}$$

After discretization, each voxel is represented by a discrete vector $\mathbf{v}^d = (l_1, l_2, ..., l_m)$.

*Difference-aware discretization.* We discretize voxel stress differences into degraded or optimized labels based on whether the difference is positive or negative. Given the stress difference $\Delta V_i(x,y,z)$ at voxel $(x,y,z)$ in the $i$-th alternative, we assign:

$$l_i = \begin{cases} -1, & \Delta V_i(x,y,z) < 0 \quad \text{/* optimized */} \\ 1, & \Delta V_i(x,y,z) \geq 0 \quad \text{/* degraded */} \end{cases}$$

After this process, each voxel is represented by a discretized vector $\Delta\mathbf{v}^d = (l_1, l_2, ..., l_m)$

Second, a recursive **region-growing** algorithm is employed, considering both spatial proximity and consistency in stress-related features of voxels. The algorithm begins by scanning voxels in lexicographic $(x,y,z)$ order to find the first unlabeled voxel, which is assigned a unique label to initiate a new segment. Next, the algorithm performs a queue-based breadth-first search to iteratively examine each neighboring voxel. A neighbor is added to the current segment only if it is unlabeled and its inclusion maintains the consistency of the segment. A segment is considered consistent if all discretized vectors within the segment are the same.

Although perfect in theory, the consistency condition is too strict, making the region-growing process easily terminated, leading to fragmented and overly fine segments. To improve robustness, we introduce an intermediate range, in which the voxel stress values or voxel differences are relabeled as 0, representing an intermediate state. The intermediate range for stress-aware discretization is $[\lambda_s - 0.02\times \text{FS}, \lambda_s + 0.02\times \text{FS}]$, where FS denotes the failure stress value. The intermediate range for difference-aware discretization is $[-0.02\times \text{FS}, 0.02\times \text{FS}]$. In this way, slight errors are allowed, for example, "optimized" voxels and "no difference" voxels can be in the same segment, making the region easier to grow. The revised consistency condition is as follows: a segment is considered consistent if in each dimension (i.e., for each design alternative), the label difference between all discretized vectors within the segment does not exceed 1. For the pseudocode, please refer to Appendix B. In addition, in stress-aware segmentation, there may be some voxels that do not exist in certain alternatives. These non-existent voxels do not affect our goal of obtaining segments with stress similarity and thus are labeled as 0 to promote the region-growing process. We have also provided a segment ranking feature based on two key metrics: variance and magnitude. Please refer to Sec. 4.5.

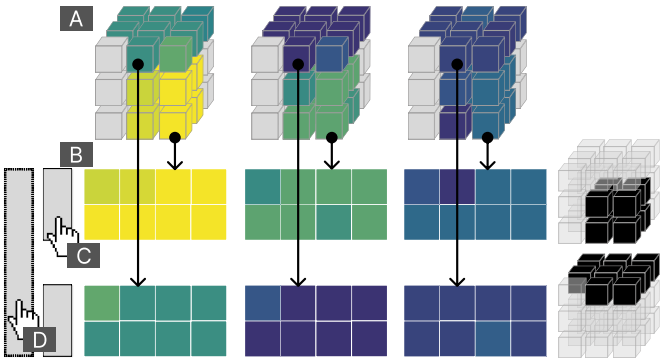


Fig. 3: Design illustration of the comparison view. (A) Three alternatives are placed at the top as column headers with stress-aware volume visualization. (B) In a matrix-based layout, each column is an alternative, each row is a segment, and each cell visualizes the segment's stress in an alternative. Within each cell, the voxels in 3D space are mapped onto pixels in 2D space with the same color but without transparency. (C) The user selects a segment and (D) groups segments by clicking the button.

#### 4.4.2 Visualization

Following *Scan sequentially* strategy [19], we design an alternative–segment matrix for both alternative-to-alternative and segment-to-segment comparisons.

**Matrix layout.** Each column represents an alternative, while each row corresponds to a distinct segment, as shown in Figure 3. To preserve the geometric context, a segment snapshot is appended to the end of each row, highlighting the segment's spatial location within the alternative model. Each cell encodes the stress data, either the stress value or stress difference, of the corresponding segment within an alternative. This layout enables the simultaneous comparison of multiple segments and alternatives in a well-organized 2D space, mitigating visual occlusion issues inherent in 3D representations.

**Voxel-to-Pixel mapping.** To visualize voxel data within a segment, we project the 3D voxel grid into a 2D layout, mapping each voxel to a pixel within the corresponding matrix cell, as illustrated in Figure 3A and B. Given two segments and three alternatives (Figure 3A), we extract voxel values at the same spatial coordinates across alternatives and assign them to the corresponding 2D position in each cell. Each pixel retains its original voxel value, using the same color scheme in the tree view but with full opacity, as shown in Figure 3B. Notably, nonexistent voxels generated by the stress-aware segmentation are encoded in light gray to visually indicate their absence within the segment in a given alternative. For example, as shown in Figure 6B1, the pixels corresponding to Models $A_{C^\# E^\#}$ and $A_{C^\#}$ appear gray, indicating nonexistent voxels in these models. In contrast, the same pixels appear green in Models $A_{B^\# D^\#}$ and $A_{C^\#}$, suggesting that this region exists only in these two alternatives and exhibits a moderate stress level. Since the column width is fixed, the row height encodes the number of voxels in a segment. The number of voxels may vary significantly between segments, resulting in cells that are either excessively long or too short—both of which hinder effective analysis. Hence, we apply a logarithmic transformation followed by min-max normalization, mapping cell heights to a perceptually balanced range, ensuring all segments remain visually accessible.

Voxel values are spatially continuous in 3D but become discontinuous when projected into 2D, leading to a visually noisy appearance. To enhance visual continuity, we employ a pixel ordering strategy based on principal component analysis (PCA). Specifically, we apply the principal components analysis to voxel vectors within each segment, obtaining the one-dimensional projection of each voxel. As voxel vectors are constructed from the stress data across alternatives, any two voxels with close projections will tend to have similar stress data, in either scheme. Ordering the pixels based on the projection can improve the visual coherence within each cell. For an explanation of our choice of PCA-based ordering and a comparison of different pixel ordering strategies, please refer to Appendix A.

### 4.5 User Interactions

To enhance user exploration of segments and the analysis of localized stress distributions, we implement the following interactive features.

**Segment pinning and highlighting.** In the comparison view, users can select segments by clicking the selection buttons located at the beginning of each row (Figure 3C). Once selected, the corresponding segments are pinned to the top of the matrix for improved visibility. At the same time, these segments are highlighted in the volume visualizations in the tree view, while unselected segments are visually filtered out, allowing users to focus on relevant structures.

**Manual segment grouping.** Default segmentation primarily considers spatial proximity and stress-related features but does not account for structural semantics. For instance, it may fail to classify the four legs of a chair as a single segment. We provide users with the ability to manually group segments. This operation builds upon segment pinning: once segments are pinned, users can click the grouping button located to the left of the selection buttons (Figure 3D) to combine them into a new, unified segment. This flexibility allows users to define semantically meaningful segments beyond those automatically generated.

**Segment ranking.** The comparison view provides a segment ranking feature to help users identify and prioritize segments of interest based on two key metrics: variance and magnitude. These metrics can be applied to either voxel stress values or voxel stress differences, depending on which kind is associated with each voxel.

The variance metric quantifies the fluctuation of voxel values across all alternative models, highlighting segments that exhibit substantial variations. It is computed as the statistical variance of voxel values aggregated within each segment. When applied to voxel stress values, it reveals regions with unstable stress distributions. When applied to voxel stress differences, it identifies segments where stress changes significantly across alternatives. The magnitude metric captures the overall intensity of voxel values within a segment. When applied to voxel stress values, it highlights regions experiencing high stress. When applied to voxel stress differences, it identifies segments with substantial stress variation. Since stress difference values can be either positive or negative, magnitude is particularly defined as the average absolute voxel value across all alternatives.

By integrating these two complementary metrics, the ranking function directs users' attention to segments with meaningful stress patterns, enabling more targeted structural analysis.

### 4.6 System Architecture

StressDiffVis is implemented as a web-based application adopting a decoupled front-end and back-end architecture to ensure modularity. The back-end server is built using the Python Flask framework, and is primarily responsible for volumetric data processing, model segmentation, and differential computation. In particular, model segmentation is optimized through the use of Numba, a Python library that accelerates computation-intensive loops via JIT compilation to fast machine code. To enhance computational efficiency, the server supports parallel processing for differential computation tasks. For operations such as segment selection, the system leverages the NumPy library to perform efficient voxel-level operations, including array filtering and mask-based manipulations. The front end is developed using TypeScript and the Vue 3 framework. The 3D visualization component is implemented with Three.js, which provides efficient support for DVR.

All development and testing were conducted on a desktop system running Windows 10, equipped with an Intel Core i7-13700K 3.40 GHz CPU, NVIDIA GeForce RTX 3070 (8 GB) GPU, and 64 GB of RAM.

## 5 Evaluation

The effectiveness and utility of StressDiffVis were evaluated through two case studies and interviews with E1 and E3. Both case studies followed the same procedure. First, we introduced the visual designs and interactions of StressDiffVis. Second, the expert selected a structure model to optimize it regarding the stress analysis with a series of structural modifications. AutoCAD was provided to the expert as a modeling software, allowing the expert to perform structural modifications. Abaqus was used to perform FEA, generating the nodal stress data. StressDiffVis serves as a comparative stress analysis tool to help the expert evaluate and compare the generated alternative models. After

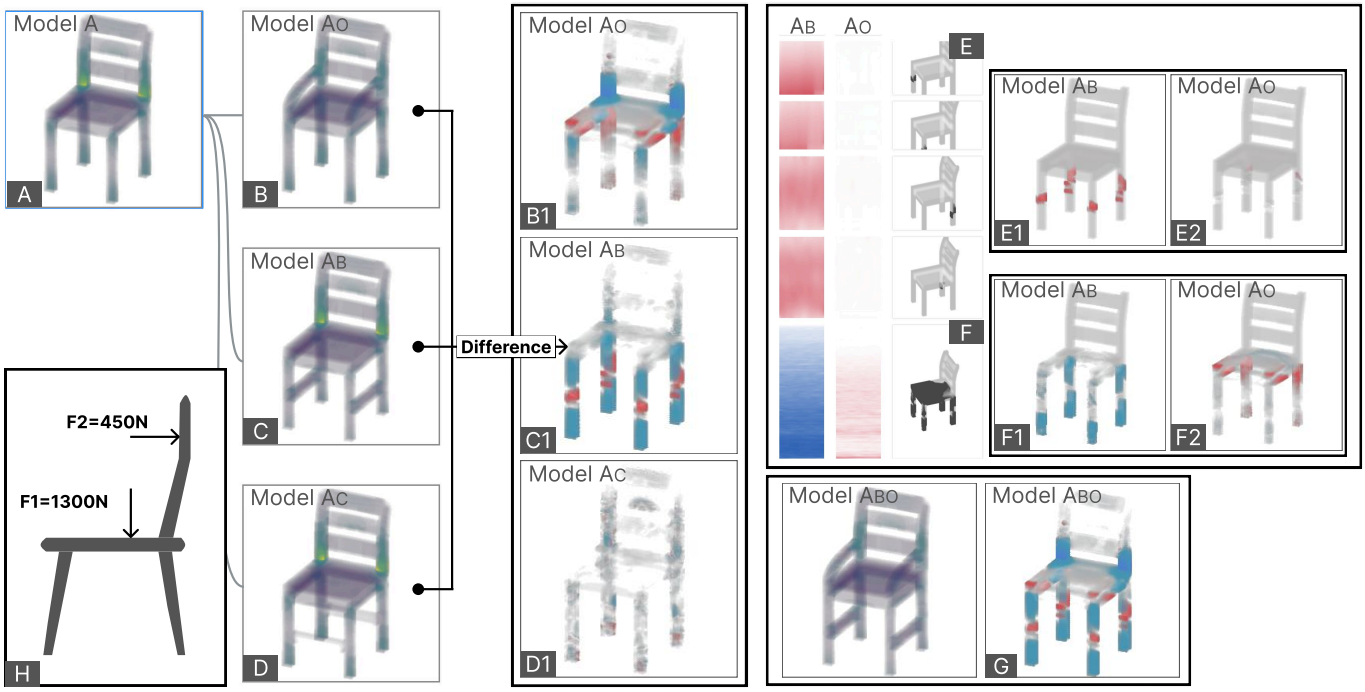


Fig. 4: Case study regarding a wooden chair. (A, B, C, and D) Stress-aware volume visualizations of Models A, $A_O$, $A_B$, and $A_C$, respectively. (B1, C1, and D1) Difference-aware volume visualizations to highlight deterioration or improvements of (B), (C), and (D), compared to (A). (E) The top four segments, corresponding to the middle section of the chair legs, are identified as regions with deterioration in Model $A_B$. (E1 and E2) Difference-aware volume visualizations of Models $A_B$ and $A_O$, compared to Model A, with filtering to include only the top four segments. (F) The bottom segment is improved in Model $A_B$ but deteriorates in Model $A_O$. (F1 and F2) Difference-aware volume visualizations of Models $A_B$ and $A_O$, compared to Model A, with filtering to include only the bottom segment. (G) Difference-aware volume visualization of Model $A_{BO}$, compared to Model A. (H) Loading conditions applied to the wooden chair.

each case study, we conducted an interview to collect expert feedback regarding StressDiffVis's practicality and effectiveness.

For better illustration, we use "Model" followed by a letter to distinguish different models, such as "Model A" and "Model B." To indicate additional modifications, we adopt subscripted notation. For example, "Model $A_C$" represents a variant of "Model A" with modification C added. In this way, "Model A" can be considered as removing modification C from "Model $A_C$."

### 5.1 Case I: Comparative Stress Analysis of a Wooden Chair

The first case study focused on a wooden chair provided by E3. The objective was to optimize the chair's structure by analyzing its stress distribution, identifying critical high-stress regions, and iteratively refining the design for structural reinforcement.

**Parameter setting.** The wooden chair was constructed from spruce, which has a density of 500 kg/m$^3$. As wood is an orthotropic material, its mechanical properties vary significantly along different anatomical directions. However, following [23], to simplify the stress analysis and facilitate efficient computation, an isotropic approximation was employed. The elastic modulus was set to $E = 3.98 \times 10^9$ Pa and Poisson's ratio to $\nu = 0.192$. While this approximation introduces simplifications and may deviate from the precise stress behavior of anisotropic materials, it preserves the relative spatial distribution and trend of stress concentrations, which are the primary focus of our visual analysis. Such trends remain meaningful for comparative purposes, especially in early-stage design evaluations. The failure stress value is set as $1.2 \times 10^7$ Pa. For loading conditions, a maximum occupant weight of 130 kg was considered to account for overweight individuals. In addition, a horizontal force of 450 N was accordingly applied to the backrest. The direction, placement, and magnitude of the applied forces were illustrated in Figure 4H.

**Overview.** After importing Model A (the base model) and loading the corresponding stress data, E3 obtained the initial visualization (Figure 4A). Model A was rendered in the tree view using volume rendering, providing an overview of the stress distribution. E3 observed that the seat of the chair appeared blue and nearly transparent, indicating minimal stress in this region. In contrast, significant stress was identified at the backrest of the chair and at the ends of the legs (i.e., the feet and hinges). Since E3 was primarily concerned with high-stress regions, low-stress regions were deemed less relevant for further analysis. E3 decided to optimize the structural design of the chair's backrest and legs to achieve a more balanced stress distribution and enhance overall structural integrity.

**Reinforcing chair backrest.** To reduce the stress of the chair's backrest, E3 introduced two armrests to Model A, resulting in Model $A_O$. A preliminary observation in Model $A_O$ was a lighter color in the backrest of the chair compared to Model A, suggesting a stress reduction (Figure 4B). However, the extent of the optimized region was still hard to recognize. Consequently, E3 selected Model A as the reference model for further comparative analysis. The system computed voxel-wise stress differences between Model A and Model $A_O$. The resulting difference visualization (Figure 4B1) indicated an extensive region of blue difference voxels, confirming a significant improvement on the backrest. Moreover, the extent of the optimized region was identifiable, extending from the bottom of the backrest to both ends of the lower rail. Based on these findings, Model $A_O$ effectively met E3's requirements for optimizing the chair's backrest.

**Reinforcing chair legs.** To optimize the stress distribution of the chair legs, E3 designed two alternatives, Models $A_B$ and $A_C$, by incorporating additional stretchers connecting the front-back chair legs and left-right chair legs, respectively. Similarly, E3 imported Models $A_B$ and $A_C$ and their corresponding stress data into the system, as shown in Figure 4C and D. However, the precise effect of the additional stretchers in Models $A_B$ and $A_C$ remained unclear. Consequently, E3 selected Model A as the reference model, and the differential computation was performed. The resulting difference visualization in the tree view (Figure 4C1 and D1) highlighted a blue region at the hinge and foot of Model $A_B$, indicating stress reduction in these regions. This corresponds to the critical high-stress regions in the chair legs of Model A that required structural improvement. In addition, stress increased in the middle section of the chair legs in Model $A_B$ (Figure 4C1). This suggests that the stretchers connecting the front-back legs effectively

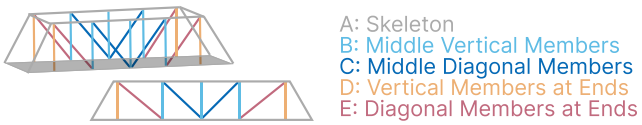


Fig. 5: Sketch of a truss bridge and notations used for describing it.

redistribute the excessive stress in the chair legs. This redistribution allowed forces to be more evenly distributed across the chair legs, improving the structural integrity.

In contrast, Model $A_C$ exhibited minimal difference compared to Model A (Figure 4D1). This outcome stems from the applied loads, including vertical forces from the seated user and horizontal forces on the chair back. The front-back stretchers in Model $A_B$ effectively resisted horizontal loads, whereas the left-right stretchers in Model $A_C$ offered no benefit under these loading conditions. The difference was also slighter compared to the differences shown in Figure 4B1 and C1. As a result, Model $A_C$ was excluded from further analysis, and Model $A_B$ was retained for the structural improvement of the chair legs.

**Combination of reinforcement.** Based on the above analysis, E3 concluded that Models $A_B$ and $A_O$ effectively optimized the chair legs and backrest, respectively. However, upon further inspection in the comparison view (Figure 4E and F), E3 observed two distinct kinds of segments, each exhibiting opposite effects.

Within the top four segments (Figure 4E), the difference voxels in Model $A_B$ appeared predominantly red, whereas those in Model $A_O$ were mostly white. This indicated that Model $A_B$ contributed to stress deterioration in this region, while Model $A_O$ had minimal impact. E3 selected the top four segments shown in Figure 4E as a segment group. By activating this group and filtering its voxels in the tree view, E3 obtained the affected region (Figure 4E1 and E2) and confirmed that the region of deterioration in model $A_B$ corresponded to the middle section of the chair legs, aligning with the previous analysis.

In contrast, in the lower segment (Figure 4F), the difference voxels in Model $A_B$ were predominantly blue, while those in Model $A_O$ were red, indicating that Model $A_B$ improved this region while Model $A_O$ worsened it. After filtering and rendering the voxels in this segment (Figure 4F1 and F2), E3 found that Model $A_O$ significantly deteriorated the upper section of the chair's back legs and the hinge. Since this region already exhibited high stress in Model A, further deterioration led to an even more uneven stress distribution in the chair legs.

To mitigate the negative impact of Model $A_O$, E3 designed a combined Model, $A_{BO}$, which integrates Models $A_B$ and $A_O$ by incorporating both back-end stretchers and armrests to balance their effects. E3 imported Model $A_{BO}$ into the system and used Model A as the reference to generate the difference rendering for Model $A_{BO}$, as shown in Figure 4G. The results indicated that Model $A_{BO}$ significantly improved the stress distribution in both the chair legs and backrest. However, an increase in stress was observed in the middle section of the chair legs. Since this region originally exhibited low stress in Model A, E3 considered this deterioration acceptable. Consequently, E3 achieved an optimized structural design for the wooden chair.

## 5.2 Case II: Comparative Stress Analysis of Truss Bridge

The second case study focused on the structural improvement of a truss bridge model provided by designer E1. The objective was to reduce the material cost of the model while retaining safety by preventing fracture.

**Parameter setting.** The truss bridge model was constructed from reinforced steel, with a density of 7850 kg/m$^3$, Young's modulus of $2.1 \times 10^{11}$ Pa, and a Poisson's ratio of 0.3. The failure stress value is set as $1.4 \times 10^8$Pa. The applied load consisted of a uniformly distributed vertical force of $4.5 \times 10^6$ N acting perpendicular to the bridge deck, in addition to the gravitational load.

**Model introduction.** During the design process, E1 tried to modify or remove some of the supports to reduce the material. For illustration, we introduce the following notations (Figure 5). The base model is denoted as $A_{B^\# C^\# E^\# D^\#}$. The letter A represents the skeleton of the truss bridge, and the letters B, C, E, and D represent the support members. Hereafter, we refer to the support member as the support for brevity. The symbol # represents the thickening of the supports. For example, $B^\#$ denotes the supports thickened from B. B denotes the three middle

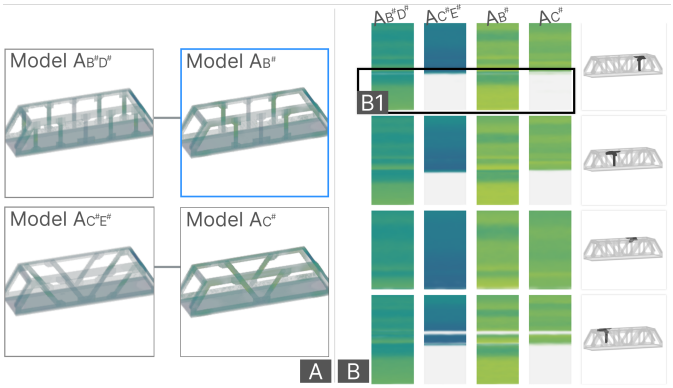


Fig. 6: Case study of a truss bridge. (A) Stress-aware volume visualization of Models $A_{B^\# D^\#}$, $A_{C^\# E^\#}$, $A_{B^\#}$, and $A_{C^\#}$, where stress variants are difficult to distinguish. (B) In the segments corresponding to connections between supports and upper chord beams, Models $A_{B^\#}$ and $A_{C^\#}$ exhibit stress deterioration. (B1) Voxels are absent in Models $A_{C^\# E^\#}$ and $A_{C^\#}$ (gray), but present with moderate stress in $A_{B^\# D^\#}$ and $A_{B^\#}$ (green).

vertical supports. C denotes the two middle diagonal supports. D denotes the two vertical supports, and E denotes the two diagonal supports, both located at the far left and right ends of the bridge.

**Removing supports V.S. thinning supports.** Initially, E1 designed three alternatives, namely, Models $A_{BCED}$, $A_{B^\# D^\#}$, and $A_{C^\# E^\#}$ (Figure 1A). In the first one, all supports were thinned, while in the latter two, diagonal or vertical supports were selectively removed, respectively. In the stress visualization of Model $A_{BCED}$, E1 observed distinct yellow regions in the diagonal supports E, indicating significant stress concentration (Figure 1B). In the other alternatives, no clear stress concentration patterns were observed.

To examine stress distribution changes, E1 selected the base model as the reference to compute stress differences and generate difference-aware visualizations (Figure 1C). First, in Model $A_{BCED}$, a significant stress increase was observed in the diagonal supports, whereas the vertical supports exhibited minimal differences (Figure 1C1). The reduced thickness of vertical supports (i.e., supports B and D) has a negligible impact on stress distribution. E1 confirmed that diagonal supports experience greater horizontal shear than vertical ones. Second, in Model $A_{B^\# D^\#}$, where diagonal supports are removed, substantial stress concentrations emerge in the vertical supports and the lower chord beam (Figure 1C2), indicating a loss of structural integrity. Third, comparing Model $A_{C^\# E^\#}$ to the base model reveals that removing the vertical supports increases stress in the lower regions of both the diagonal supports and the lower chord beam (Figure 1C3).

To more clearly analyze vertical and diagonal supports, E1 turned to the comparison view, where the alternatives were segmented based on stress distribution. Based on the performance of the segmentation results, E1 selected $\lambda_s = 0.57 \times FS$ as the stress threshold. By ranking segments according to their variances, E1 noticed the segments corresponding to the vertical supports (Figure 1D). Although the stress largely increased in Model $A_{B^\# D^\#}$, the magnitude remained within an acceptable range, as indicated by the green color. E1 then ranked the segments based on their magnitude. Segments exhibiting notably high stress levels within the diagonal support regions were identified (Figure 1E). E1 then grouped these segments together and pinned them at the top. Another pattern emerged clearly: in Model $A_{BCED}$, these segments appeared yellow with high stress.

Notably, among all alternatives, only this model exhibited such high stress in these regions, reinforcing previous findings. To further investigate this stress concentration, we examined the corresponding pixel positions in other alternatives. Specifically, in Models $A_{B^\# C^\# E^\# D^\#}$ and $A_{C^\# E^\#}$, these regions consistently appeared green, indicating low stress. This suggests that the removal of the thickened vertical support in Model $A_{B^\# C^\# E^\# D^\#}$ does not lead to stress increase in the diagonal support. Consequently, the thickened vertical support in Model $A_{B^\# C^\# E^\# D^\#}$ may be considered removable.

E1 retained Models $A_{B^\# D^\#}$ and $A_{C^\# E^\#}$, as they reduced material costs while exhibiting an acceptable stress distribution. In contrast, Model $A_{BCED}$ was discarded due to its high-stress concentration.

**Thinning supports further.** E1 further explored whether the bridge's material could be further reduced. E1 removed the supports at the bridge ends from Models $A_{B^\# D^\#}$ and $A_{C^\# E^\#}$, respectively, obtaining Models $A_{B^\#}$ and $A_{C^\#}$. E1 wanted to check whether either of these models was available. E1 first encountered difficulty in figuring out how stress increases compared to the previous models (Figure 6A). Thus, E1 turned to analyze the segments. By ranking the segments according to their variances, E1 identified the segments where the stress of Models $A_{B^\#}$ and $A_{C^\#}$ largely increased compared to Models $A_{B^\# D^\#}$ and $A_{C^\# E^\#}$. For example, shown in Figure 6B, the connections of the supports and the upper chord beam in both Models $A_{B^\#}$ and $A_{C^\#}$ exhibited obvious stress concentration. Stress concentration at the connection is more dangerous than that at the supports, as connections are fatigue-prone parts. Finally, E1 believed further reduction of materials is undesirable and decided to test Models $A_{B^\# D^\#}$ and $A_{C^\# E^\#}$ under other load conditions.

### 5.3 Expert Interviews

We conducted one-on-one interviews with domain experts after the case studies to gather feedback on StressDiffVis.

**Effectiveness.** Experts acknowledged the practical value of StressDiffVis in facilitating stress comparison and guiding structural improvements, highlighting its potential impact on engineering workflows. E3 noted that the system effectively revealed stress differences between structures, stating, *"It saves me from playing 'spot the difference' between stress plots."* Furthermore, both experts confirmed that the multi-alternative comparative visualization provides a capability that is currently lacking in CAE post-processing tools.

**Visualization.** They appreciated StressDiffVis's visual design. Following a brief introduction, they readily understood the visual design. E3 highlighted the volume visualization alleviates occlusion issues common in contour plots, reducing the need for continuous camera manipulation. E1 emphasized the utility of the alternative-segment matrix, along with model segmentation, in facilitating localized, occlusion-free comparisons. He remarked, *"The matrix allows me to efficiently detect localized stress variations across multiple alternatives."*

**Suggestions.** Experts suggested the inclusion of numerical stress information. E3 acknowledged that while the current color and opacity encodings effectively differentiate between high- and low-stress regions, as well as stress variations, incorporating numerical stress values would enhance his ability to assess improvements and determine whether further refinements are warranted. E1 similarly emphasized the importance of numerical data. He noted that although some regions may exhibit visible stress deterioration, their actual stress values might remain low and non-critical. To address this, he suggested introducing an option to filter out low-stress regions, thereby emphasizing regions where high stress has worsened.

## 6 Discussion

This section discusses the implications, scalability, generalizability, and limitations of StressDiffVis, and potential directions for future research.

**Implications.** To the best of our knowledge, StressDiffVis is the first visual analytics approach for comparative stress analysis across multiple design alternatives. It addresses a key limitation of existing stress analysis platforms, which primarily evaluate models side-by-side and lack support for systematic stress distribution comparisons during design iterations. By combining volumetric representation with model segmentation, StressDiffVis enables engineers to explore stress distributions and variations between alternatives from both localized and global perspectives, offering deeper insights into structural behavior and facilitating more informed decision-making.

**Scalability.** While real-time rendering of complex, large-scale structural models is beyond the scope of this work, StressDiffVis is designed to integrate seamlessly into engineering workflows. In the structural analysis of a large-scale model, for example, an entire automobile, engineers typically focus on critical substructures (e.g., the chassis and suspension) rather than analyzing the entire model at full resolution. To accommodate this workflow, only key structures are imported into StressDiffVis, ensuring efficient exploration. Furthermore, the model segmentation process is optimized for interactive use through Numba JIT compilation. In Case I, segmentation is completed within 13 seconds for five chair alternatives, each with over 1.46 million voxels. In Case II, segmentation is completed within 30 seconds for six truss bridge alternatives, each with over 4.2 million voxels. These results demonstrate that StressDiffVis maintains acceptable responsiveness across models of varying complexity, making it suitable for practical engineering analysis.

**Generalizability.** The proposed visual analytics pipeline, which encompasses volumetric transformation, differential computation, and model segmentation, is inherently independent of specific simulation software or geometric models. Thus, StressDiffVis demonstrates high generalizability across various structural design scenarios and model types. It can be seamlessly integrated into existing engineering workflows, supporting comparative stress analysis across diverse structural designs without requiring significant modifications.

**Lesson Learned.** We want to share two lessons for the visualization community. On the technical side, our work demonstrates that CAE simulation data can be structured into voxel-based representations to support localized difference analysis. This strategy is applicable to other domains, such as fluid dynamics, thermal analysis, and electromagnetic field studies, where comparison across model variants is equally critical. On the methodological side, we observe that while many industrial tools have matured over decades, there remain untapped opportunities for visual analytics to enhance their effectiveness. We encourage researchers to explore such under-addressed areas where visualization can bring new value.

**Limitations and Future Work.** Despite its effectiveness, StressDiffVis also faces limitations, revealing potential areas for future research.

*Directional Stress Analysis.* This study focuses on scalar stress measures, which limits its applicability in scenarios where directional effects are critical. As part of future work, we aim to support full stress tensor analysis and anisotropic material modeling. Visualizing and comparing tensor fields introduces challenges, requiring new encoding strategies and interaction techniques. Potential directions include the use of tensor invariants, principal stress directions, or interactive projection methods to enable interpretable, comparative visualization of directional stress patterns across design alternatives.

*Model segmentation.* The model segmentation in StressDiffVis primarily focuses on voxel-level stress similarity and spatial continuity without considering structural semantics. As a result, semantically related but spatially disconnected components—such as symmetrical elements (e.g., chair legs)—are treated as independent segments. This may lead to redundancy and fragmented segments, potentially complicating interpretation. Future work could explore the integration of semantic-aware segmentation. By leveraging advanced 3D computer vision models [46] or geometry-based models [15], semantically related segments could be merged, improving the clarity and interpretability of the analysis. In addition, the segmentation currently uses a default lexicographic scan to initialize region growing. Allowing user-defined seeding could offer more flexibility in guiding segmentation, and we consider this a potential future extension.

*Virtual/Augmented Reality.* Virtual Reality (VR) and Augmented Reality (AR) technologies provide immersive perception and intuitive interaction, enhancing users' spatial awareness and efficiency in complex data analysis [18, 47, 51, 54]. To the best of our knowledge, the application of VR/AR for stress analysis remains unexplored. We plan to extend StressDiffVis into a VR/AR environment to further support immersive and interactive stress analysis.

## 7 Conclusion

This study presents StressDiffVis, a visual analytics approach for comparing stress distributions across multiple design alternatives. Developed through expert-informed collaboration, the system addresses key requirements in stress analysis and comparative evaluation. We employ a volumetric representation to enhance the visibility of stress distributions and leverage voxel-wise differential computation to visualize stress differences. To support localized stress analysis, we utilize stress-based model segmentation. We also organize alternatives and their segments in a matrix layout, where the volumetric data of each segment is mapped into a 2D pixel representation, enabling occlusion-free, inter-model and intra-model stress comparison. Two real-world case studies and expert interviews demonstrate the system's effectiveness in structural design analysis.

## Acknowledgments

The authors wish to thank all reviewers for their constructive comments. The work was supported by the National Key R&D Program of China (2023YFB3309100), the National Natural Science Foundation of China (62402184), and the Fundamental Research Funds for the Central Universities, SCUT (x2rjD2250190).

## References

[1] E. Bradner, F. Iorio, M. Davis, et al. Parameters tell the design story: ideation and abstraction in design optimization. In *Proceedings of the Symposium on Simulation for Architecture & Urban Design*, vol. 26, pp. 1–8. Citeseer, 2014. 1

[2] E. Carra and F. Pellacini. Scenegit: A practical system for diffing and merging 3D environments. *ACM Transactions on Graphics*, 38(6):159:1–159:15, 2019. doi: 10.1145/3355089.3356550 2

[3] X. Chen, C. Zheng, and K. Zhou. Example-based subspace stress analysis for interactive shape design. *IEEE Transactions on Visualization and Computer Graphics*, 23(10):2314–2327, 2017. doi: 10.1109/TVCG.2016.2618875 2

[4] T. Delmarcelle and L. Hesselink. Visualizing second-order tensor fields with hyperstreamlines. *IEEE Computer Graphics and Applications*, 13(4):25–33, 1993. doi: 10.1109/38.219447 2

[5] F. Demarco, F. Bertacchini, C. Scuro, E. Bilotta, and P. Pantano. The development and application of an optimization tool in industrial design. *International Journal on Interactive Design and Manufacturing*, 14(3):955–970, 2020. doi: 10.1007/s12008-020-00679-4 1, 2

[6] I. Demir, C. Dick, and R. Westermann. Multi-charts for comparative 3D ensemble visualization. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2694–2703, 2014. doi: 10.1109/TVCG.2014.2346448 2, 3

[7] Z. Deng, H. Chen, Q.-L. Lu, Z. Su, T. Schreck, J. Bao, and Y. Cai. Visual comparative analytics of multimodal transportation. *Visual Informatics*, 9(1):18–30, 2025. doi: 10.1016/j.visinf.2025.01.001 2

[8] Z. Deng, J. Huang, C. Ruan, J. Li, S. Gao, and Y. Cai. Volume-based space-time cube for large-scale continuous spatial time series. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–15, 2025. doi: 10.1109/TVCG.2025.3537115 2

[9] Z. Deng, Y. Liu, M. Zhu, D. Xiang, H. Yu, Z. Su, Q. Lu, T. Schreck, and Y. Cai. Trasculptor: Visual analytics for enhanced decision-making in road traffic planning. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–16, 2025. doi: 10.1109/TVCG.2025.3532498 4

[10] J. D. Denning and F. Pellacini. MeshGit: Diffing and merging meshes for polygonal modeling. *ACM Transactions on Graphics*, 32(4):1–10, 2013. doi: 10.1145/2461912.2461942 2

[11] C. Dick, J. Georgii, R. Burgkart, and R. Westermann. Stress tensor field visualization for implant planning in orthopedics. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1399–1406, 2009. doi: 10.1109/TVCG.2009.184 2

[12] C. Dick, J. Georgii, R. Burgkart, and R. Westermann. A 3D simulation system for hip joint replacement planning. In *Proceedings of the World Congress on Medical Physics and Biomedical Engineering*, pp. 363–366. Springer Berlin Heidelberg, 2010. doi: 10.1007/978-3-642-03882-2_96 2

[13] J. Dobos, C. Fan, S. Friston, and C. Wong. Screen space 3D diff: A fast and reliable method for real-time 3D differencing on the web. In *Proceedings of the International ACM Conference on 3D Web Technology*, pp. 9:1–9:9. ACM, 2018. doi: 10.1145/3208806.3208809 2

[14] J. Dobos and A. Steed. 3D Diff: An interactive approach to mesh differencing and conflict resolution. In *Proceedings of SIGGRAPH Asia Technical Briefs*, pp. 20:1–20:4. ACM, 2012. doi: 10.1145/2407746.2407766 2

[15] Q. Dong, Z. Wang, M. Li, J. Gao, S. Chen, Z. Shu, S. Xin, C. Tu, and W. Wang. Laplacian2Mesh: Laplacian-based mesh understanding. *IEEE Transactions on Visualization and Computer Graphics*, 30(7):4349–4361, 2024. doi: 10.1109/TVCG.2023.3259044 9

[16] F. Flager and J. Haymaker. A comparison of multidisciplinary design, analysis and optimization processes in the building construction and aerospace industries. In *Proceedings of the International Conference on Information Technology in Construction*, pp. 625–630, 2007. 2

[17] F. G. Fuchs and J. M. Hjelmervik. Interactive isogeometric volume visualization with pixel-accurate geometry. *IEEE Transactions on Visualization and Computer Graphics*, 22(2):1102–1114, 2016. doi: 10.1109/TVCG.2015.2430337 2

[18] A. Gall, E. Gröller, and C. Heinzl. Imndt: Immersive workspace for the analysis of multidimensional material data from non-destructive testing. In *Proceedings of the 27th ACM Symposium on Virtual Reality Software and Technology*, article no. 9, 2021. doi: 10.1145/3489849.3489851 9

[19] M. Gleicher. Considerations for visualizing comparison. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):413–423, 2018. doi: 10.1109/TVCG.2017.2744199 3, 5, 6

[20] J. C. Hart. Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces. *The Visual Computer*, 12(10):527–545, 1996. 5

[21] J. Heer, J. D. Mackinlay, C. Stolte, and M. Agrawala. Graphical histories for visualization: Supporting analysis, communication, and evaluation. *IEEE Transactions on Visualization Computer Graphics*, 14(6):1189–1196, 2008. doi: 10.1109/TVCG.2008.137 4

[22] C. Hergl, C. Blecha, V. Kretzschmar, F. Raith, F. Günther, M. Stommel, J. Jankowai, I. Hotz, T. Nagel, and G. Scheuermann. Visualization of tensor fields in mechanics. *Computer Graphics Forum*, 40(6):135–161, 2021. doi: 10.1111/CGF.14209 2

[23] I. Horman, S. Hajdarević, S. Martinović, and N. Vukas. Numerical analysis of stress and strain in a wooden chair. *Wood Industry/Drvna Industrija*, 61(3), 2010. 7

[24] G. Kindlmann. Superquadric tensor glyphs. In *Proceedings of the Joint Eurographics - IEEE TCVG Conference on Visualization*, 8 pages, p. 147–154, 2004. 2

[25] A. Kratz, B. Meyer, and I. Hotz. A visual approach to analysis of stress tensor fields. In *Scientific Visualization: Interactions, Features, Metaphors*, vol. 2 of *Dagstuhl Follow-Ups*, pp. 188–211. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, 2011. doi: 10.4230/DFU.VOL2.SCIVIZ.2011.188 2

[26] A. Kratz, M. Schöneich, V. Zobel, B. Burgeth, G. Scheuermann, I. Hotz, and M. Stommel. Tensor visualization driven mechanical component design. In *IEEE Pacific Visualization Symposium, PacificVis 2014*, pp. 145–152. IEEE Computer Society, 2014. doi: 10.1109/PACIFICVIS.2014.51 2

[27] K. Lu, Y. Chen, Z. Luo, Y. Dong, and C. Lv. Visual comparison of hierarchies with inherited attributes: application to multi-regional mrl standards in food safety. *Journal of Visualization*, 28(3):553–568, 2025. 2

[28] M. M. Malik, C. Heinzl, and M. E. Gröller. Comparative visualization for parameter studies of dataset series. *IEEE Transactions on Visualization and Computer Graphics*, 16(5):829–840, 2010. doi: 10.1109/TVCG.2010.20 2, 3

[29] J. Matejka, M. Glueck, E. Bradner, A. Hashemi, T. Grossman, and G. W. Fitzmaurice. Dream lens: Exploration and visualization of large-scale generative design datasets. In *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems*, p. 369, 2018. doi: 10.1145/3173574.3173943 2

[30] M. Meuschke, S. Voß, O. Beuing, B. Preim, and K. Lawonn. Glyph-based comparative stress tensor visualization in cerebral aneurysms. *Computer Graphics Forum*, 36(3):99–108, 2017. doi: 10.1111/CGF.13171 2, 3

[31] R. v. Mises. Mechanik der festen körper im plastisch- deformablen zustand. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, 1913:582–592, 1913. 2

[32] J. Mountstephens and J. Teo. Progress and challenges in generative product design: A review of systems. *Computers*, 9(4):80, 2020. doi: 10.3390/computers9040080 1

[33] A. Narechania, S. Guo, E. Koh, A. Endert, and J. Hoffswell. Utilizing provenance as an attribute for visual data analysis: A design probe with provenancelens. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–14, 2025. doi: 10.1109/TVCG.2025.3571708 4

[34] T. Oster, C. Rössl, and H. Theisel. Core lines in 3D second-order tensor fields. *Computer Graphics Forum*, 37(3):327–337, 2018. doi: 10.1111/cgf.13423 2

[35] J. Panetta, A. Rahimian, and D. Zorin. Worst-case stress relief for microstructures. *ACM Transactions on Graphics*, 36(4):1–16, 2017. doi: 10.1145/3072959.3073649 2

[36] M. Patel and D. H. Laidlaw. Visualization of 3D stress tensor fields using superquadric glyphs on displacement streamlines. *IEEE Transactions on Visualization and Computer Graphics*, 27(7):3264–3276, 2021. doi: 10.1109/TVCG.2020.2968911 2

[37] J. Schmidt, R. Preiner, T. Auzinger, M. Wimmer, M. E. Gröller, and S. Bruckner. YMCA - Your mesh comparison application. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology*, pp. 153–162, 2014. doi: 10.1109/VAST.2014.7042491 2

[38] T. Schultz and G. L. Kindlmann. Superquadric glyphs for symmetric second-order tensors. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1595–1604, 2010. doi: 10.1109/TVCG.2010.199 2

[39] C. T. Silva, J. Freire, and S. P. Callahan. Provenance for visualizations: Reproducibility and beyond. *Computing in Science & Engineering*, 9(5):82–89, 2007. doi: 10.1109/MCSE.2007.106 4

[40] V. Singh and N. Gu. Towards an integrated generative design framework. *Design studies*, 33(2):185–207, 2012. doi: 10.1016/j.destud.2011.06.001 1, 2

[41] O. Stava, J. Vanek, B. Benes, N. Carr, and R. Měch. Stress relief: Improving structural strength of 3D printable objects. *ACM Transactions on Graphics*, 31(4):1–11, 2012. doi: 10.1145/2185520.218554 2

[42] R. van Pelt, R. Gasteiger, K. Lawonn, M. Meuschke, and B. Preim. Comparative blood flow visualization for cerebral aneurysm treatment assessment. *Computer Graphics Forum*, 33(3):131–140, 2014. doi: 10.1111/CGF.12369 2, 3

[43] J. Wang, C. Neuhauser, J. Wu, X. Gao, and R. Westermann. 3D-TSV: The 3D trajectory-based stress visualizer. *Advances in Engineering Software*, 170:103144, 2022. doi: 10.1016/J.ADVENGSOFT.2022.103144 2

[44] Q. Wang, R. Sheng, S. Ruan, X. Jin, C. Shi, and M. Zhu. Synthlens: Visual analytics for facilitating multi-step synthetic route design. *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–14, 2025. doi: 10.1109/TVCG.2025.3552134 2

[45] W. Wang, T. Y. Wang, Z. Yang, L. Liu, X. Tong, W. Tong, J. Deng, F. Chen, and X. Liu. Cost-effective printing of 3D objects with skin-frame structures. *ACM Transactions on Graphics*, 32(6):1–10, 2013. doi: 10.1145/2508363.2508382 2

[46] Z. Wang and F. Lu. VoxSegNet: Volumetric cnns for semantic part segmentation of 3d shapes. *IEEE Transactions on Visualization and Computer Graphics*, 26(9):2919–2930, 2020. doi: 10.1109/TVCG.2019.2896310 9

[47] Z. Wang, Q. Zhao, Y. Zhang, J. Zhang, G. Shan, X. Zhou, and D. Tian. Clayvolume: A progressive refinement interaction system for immersive visualization. *Visual Informatics*, 9(1):71–83, 2025. doi: 10.1016/j.visinf.2025.01.003 9

[48] G. Wei, L. Ma, Y. Zhou, C. Wang, J. Zheng, and Y. He. Design and optimization of self-supporting surfaces with arch beams. *IEEE Transactions on Visualization and Computer Graphics*, 31(7):4003–4017, 2025. doi: 10.1109/TVCG.2024.3388507 2

[49] J. Weissenbock, B. Fröhler, M. E. Gröller, J. Kastner, and C. Heinzl. Dynamic volume lines: Visual comparison of 3D volumes through space-filling curves. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):1040–1049, 2019. doi: 10.1109/TVCG.2018.2864510 3

[50] K. Xu, A. Ottley, C. Walchshofer, M. Streit, R. Chang, and J. Wenskovitch. Survey on the analysis of user interactions and visualization provenance. *Computer Graphics Forum*, 39(3):757–783, 2020. doi: 10.1111/cgf.14035 4

[51] T. Xu, X. Ren, J. Yang, B. Sheng, and E. Wu. Efficient binocular rendering of volumetric density fields with coupled adaptive cube-map ray marching for virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 30(10):6625–6638, 2024. doi: 10.1109/TVCG.2023.3322416 9

[52] L. Zaman, W. Stuerzlinger, C. Neugebauer, R. Woodbury, M. Elkhaldi, N. Shireen, and M. Terry. GEM-NI: A system for creating and managing alternatives in generative design. In *Proceedings of the annual ACM Conference on Human Factors in Computing Systems*, pp. 1201–1210, 2015. doi: 10.1145/2702123.270239 1, 2

[53] C. Zhang, T. Schultz, K. Lawonn, E. Eisemann, and A. Vilanova. Glyph-based comparative visualization for diffusion tensor fields. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):797–806, 2015. doi: 10.1109/TVCG.2015.2467435 2, 3

[54] Y. Zhang, Z. Wang, J. Zhang, G. Shan, and D. Tian. A survey of immersive visualization: Focus on perception and interaction. *Visual Informatics*, 7(4):22–35, 2023. doi: 10.1016/j.visinf.2023.10.003 9

[55] Q. Zhou, J. Panetta, and D. Zorin. Worst-case structural analysis. *ACM Transactions on Graphics*, 32(4):137–1, 2013. doi: 10.1145/2461912.2461967 2

[56] Q. Zou, Y. Wu, Z. Liu, W. Xu, and S. Gao. Intelligent CAD 2.0. *Visual Informatics*, 8(4):1–12, 2024. doi: 10.1016/j.visinf.2024.10.001 1