



REGULAR PAPER

Mai Dahshan · Nicholas Polys · Leanna House · Chris North · Ryan M. Pollyea · Terece L. Turton · David H. Rogers

Human–machine partnerships at the exascale: exploring simulation ensembles through image databases

Received: 5 April 2023 / Revised: 29 February 2024 / Accepted: 2 April 2024

This is a U.S. Government work and not under copyright protection in the US; foreign copyright protection may apply 2024

Abstract The explosive growth in supercomputers capacity has changed simulation paradigms. Simulations have shifted from a few lengthy ones to an ensemble of multiple simulations with varying initial conditions or input parameters. Thus, an ensemble consists of large volumes of multi-dimensional data that could go beyond the exascale boundaries. However, the disparity in growth rates between storage capabilities and computing resources results in I/O bottlenecks. This makes it impractical to utilize conventional post-processing and visualization tools for analyzing such massive simulation ensembles. In situ visualization approaches alleviate I/O constraints by saving predetermined visualizations in image databases during simulation. Nevertheless, the unavailability of output raw data restricts the flexibility of post hoc exploration of in situ approaches. Much research has been conducted to mitigate this limitation, but it falls short when it comes to simultaneously exploring and analyzing parameter and ensemble spaces. In this paper, we propose an expert-in-the-loop visual exploration analytic approach. The proposed approach leverages: feature extraction, deep learning, and human expert–AI collaboration techniques to explore and analyze image-based ensembles. Our approach utilizes local features and deep learning techniques to learn the image features of ensemble members. The extracted features are then combined with simulation input parameters and fed to the visualization pipeline for in-depth exploration and analysis using human expert + AI interaction techniques. We show the effectiveness of our approach using several scientific simulation ensembles.

Keywords Human–computer interaction (HCI) · Visual analytics · Simulation ensembles · In situ visualization · Human–AI interaction · Neural networks

1 Introduction

Simulations are used across numerous computational science domains, such as atmospheric science, computational fluid dynamics, astrophysics, and particle physics, to study challenging and complex phenomena. However, running a few long simulations do not precisely capture all features of the studied phenomena or manage uncertainty in the model, thus limiting scientists' exploration space. To avoid making erroneous decisions, a simulation ensemble is carried out using different configurations (i.e., parameter

M. Dahshan (✉)
School of Computer, University of North Florida, Jacksonville, FL, USA
E-mail: mdahshan@unf.edu

N. Polys · L. House · C. North · R. M. Pollyea
Computer Science Department, Virginia Tech, Blacksburg, VA, USA

T. L. Turton · D. H. Rogers
Los Alamos National Laboratory, Los AlamosNM, USA

Published online: 17 May 2024

settings, computational models, or boundary/initial conditions) to sample representative states occurring in the studied phenomena (Kovalchuk and Boukhanovsky 2015). Hence, the primary goals of using simulation ensembles are deepening the understanding of the simulated model, exploring parameter sensitivity, and examining patterns, relationships, and trends between and among ensemble members (Mahajan et al. 2017; Höllt et al. 2014).

The continuous increase in computing capacity has a significant driving force in addressing these complex problems, generating larger datasets, and developing more high-resolution models and simulations at exascale. This is happening in many fields, such as medical modeling, climate modeling, or fluid dynamics. These exascale simulation ensembles are modeled using high-dimensional parameters and produce massive datasets. The increasing dimensionality and complexity of ensembles usually lead to I/O and storage constraints in terms of storage capacity and transfer rates, therefore, imposing limits on the availability of full simulation raw data for post hoc analysis and exploration. In situ approaches have been developed to minimize the amount of data transferred over networks and written to disk by coupling computation and visualization tasks. In this case, visual portrayals (i.e., images) are created while the simulation results are in memory (Bauer et al. 2016).

Despite the effectiveness of in situ approaches in handling I/O and memory constraints, they still present some restrictions on the visual analysis and exploration of simulations. To address this, the Cinema framework (Ahrens et al. 2014) has been proposed to assist with the exploration of in situ visualization results. Cinema is a novel in situ approach for capturing, storing, and analyzing exascale simulation data. It saves data abstracted from various camera angles, such as images and data parameters, allowing instant access to numerous data views. While Cinema does offer support for data browsing, management, and exploration, its current exploration capabilities are primarily focused on view parameters, such as camera viewing angles and visual parameter maps. As a result, there is a need to expand Cinema's capabilities with new interactive exploration approaches that utilize human expertise during the exploration and analysis of simulation ensembles.

Exploring simulation ensemble parameter and ensemble spaces simultaneously is a challenging task due to the high complexity of relationships and associations between simulation outputs and input parameters. Many visual analysis techniques have been developed to understand and explore parameter sensitivity, optimization, uncertainty, or/and the differences and similarities between different ensemble members (Wang et al. 2018). However, the majority of these approaches rely on either aggregated solutions (i.e., descriptive statistics) (Mirzargar et al. 2014) or a sample of ensemble members to reveal the correspondence and associations between simulation inputs and outputs of spatial or spatiotemporal ensembles. These techniques assume full access to raw output data, which may not be possible for exascale image-based ensembles.

This paper proposes a visual approach to support the exploration of high-dimensional image-based parameter and ensemble spaces. Our approach is built on significantly expanding a current visualization tool, GLEE (Dahshan et al. 2020), by integrating the Cinema framework to produce Cinema-GLEE (C-GLEE). C-GLEE is different from GLEE in that it uses image databases and input parameters instead of summary statistics derived from numerical raw simulation data to analyze and explore ensembles. This calls for: 1) additional procedures to extract meaningful information from image databases, 2) merging them with simulation input parameters, and 3) a reformulation of GLEE's visualization and statistics pipeline to ensure coherence between user interactions and model specification. C-GLEE provides scientists with an intuitive, user-friendly tool for interactively browsing, analyzing, and exploring image-based simulation ensembles. In turn, scientists can recognize and understand visually complex insights patterns and structures in image-based ensembles, such as image correlations and parameter sensitivity together. In summary, the contributions of this paper are the following:

- rephrasing GLEE's statistics, visualization, and interaction pipelines to execute C-GLEE for the integration of exascale workflows.
- demonstrating that local feature extraction techniques along with deep learning (DL) models can be applied to analyze and explore image-based ensembles, while preserving sufficient information for scientists to derive meaningful scientific insights.
- applying C-GLEE to Geoscience data from in situ simulations (Cinema outputs) and supporting expert's insights into their ensemble exploration.

2 Related work

This section reviews prior research work in ensemble visualization, image-based in situ visualization, and image feature extraction.

2.1 Ensemble visualization

Ensemble visualization techniques have been introduced in a wide range of scientific domains. These techniques visualized either ensemble members, their parameters, or both simultaneously. Ensemble visualizations have been represented either as a comparative visualization of a handful of ensemble members or as an aggregation of several ensemble members (Wang et al. 2018). Most of these visualizations have focused on analyzing the variability between ensemble members using techniques such as pseudo-coloring (Hummel et al. 2013) and glyphs (Bensema et al. 2015). More complex techniques have been developed to demonstrate variations in the ensemble using isosurfaces (Ma and Entezari 2018), summary statistics (Mirzargar et al. 2014; de Souza et al. 2022), density estimation (Leistikow et al. 2020), feature bagging (Xu et al. 2018), and clustering (Kumpf et al. 2018).

Parameter space visualizations enable scientists to study the correlation between parameters, parameter sensitivity, and optimization using different methodologies, including summary statistics (Ribés et al. 2019), parallel coordinate plots (Kumpf et al. 2021), glyphs (Sanyal et al. 2010), and probabilistic features (Petz et al. 2012). Increasing in complexity, various high-dimensional parameter space visualization techniques have been developed spanning several tasks including but not limited to, comparison (Sedlmair et al. 2014), parameter sensitivity (Orban et al. 2018), optimization (Torsney-Weir et al. 2011), and prediction (He et al. 2019). On the other hand, several multi-view visualization exploring ensemble and parameter spaces simultaneously have been introduced (Luciani et al. 2018).

The ensemble visualization and visual analysis techniques mentioned earlier targeted domain-specific problems with a limited number of simulation parameters (i.e., inputs and outputs). In addition, these techniques did not incorporate human intuition or expertise as a key component of the visual analytics process, which restricted the ability to explore the data thoroughly. As the scientific community transitions to exascale simulations, more in situ visualization would be produced to overcome memory and I/O constraints. This necessitates the need for visual exploration tools to analyze and explore image-based ensembles. A recently developed visualization tool, PEViz (Zhang et al. 2022), proposed an in situ progressive method with a visual analytics system to explore ocean ensemble data. However, the proposed tool exploration capabilities did not account for parameter sensitivity nor explore correlations and relationships among ensemble members.

2.2 Image-based in situ visualization

In situ approaches have been developed to overcome storage and I/O limitations. Early implementations of in situ approaches focused on steering the simulation by coupling the visualization routine with the simulation routine. Advanced techniques have been adopted to address time-varying datasets (Fernandes et al. 2014), data compression (Di and Cappello 2016), depth map (Ye et al. 2015), pixel color (Tikhonova et al. 2010), light field (Meyer et al. 2005), and pixel rays (Shareef et al. 2006) for volume rendering. On the other hand, several approaches, such as Cinema, have been developed to explore in situ visualization. More advanced features have been added to Cinema, including but not limited to feature-centric queries (Orban et al. 2020), geometry buffers (Lukasczyk et al. 2020), zooming (Maack et al. 2020). However, these image-based approaches do not support a full semantic exploration of ensemble parameter space.

2.3 Feature extraction techniques

Feature extraction (FE) techniques are used to identify and extract relevant and meaningful features from complex high-dimensional data. These techniques reduce the dimensionality of the data while preserving the essential information contained within. The reduction of dimensions could be linear (e.g., random projection (RP) and principal component analysis (PCA)) or nonlinear (i.e., kernel PCA and autoencoders) (Kunang et al. 2018). FE techniques seek to identify the most descriptive and informative feature sets, which can be global or local feature extraction techniques. Global features' represent features extracted from the overall image (such as shape, color, or texture) (Ping Tian 2013).

Local feature extraction techniques acquire the image's low-level feature information (e.g., edges, changes in intensity, and gradients) to identify its distinctive representation. Local feature extraction techniques consist of two steps: feature detection and feature description. Feature detection extracts key points by identifying the salient areas in an image. Feature description creates a vector representing the neighborhood of the detected key point (Tuytelaars and Mikolajczyk 2008). Several approaches combining local and global features have been proposed to take advantage of both techniques. Risojević and Babić (2012) combined SIFT descriptors and Gabor filters to extract the image's local and texture features.

Recently, several DL architectures have been developed to learn representative features directly from the raw pixels of an image. He et al. (2018) extract spectral and spatial features from the image via 2D convolutional neural network. Chen et al. (2020) used a deep neural network to determine bone age by extracting features from the X-ray images. Moreover, there have been several attempts to combine global and local feature extraction with deep learning models. Gao et al. (2015) proposed DEFEATnet, a deep learning network that integrates SIFT feature extraction technique into the deep learning architecture. However, these feature extraction techniques give limited attention to scientific data, particularly in the context of simulation ensembles, where the characteristics of the simulation model vary from one simulation to another.

Our proposed approach focused on the post hoc exploration of in situ generated image databases by integrating in situ visualization with human expert–AI interaction techniques to empower the visual analysis process of image databases. The proposed approach allows scientists to explore parameter and ensemble spaces simultaneously.

3 Approach

To facilitate the exploration and analysis of high-dimensional image-based ensembles, we propose a human expert–AI partnership approach that is manifested in a visual analytics system. Cinema-GLEE (C-GLEE) is an interactive multi-view visual analytics system that helps scientists analyze, search, filter, and explore high-dimensional simulation ensembles by blending human expertise and intuition with statistics and machine learning. C-GLEE incorporates human expert–AI interaction into its visual analysis process to allow scientists to interactively explore correlations and similarities between ensemble members, exploring the influence of input parameters on simulation output. In this section, we explain the system design and provide an overview of C-GLEE, detailing its various functionalities and interactions that assist in exploring the ensemble.

3.1 System design

The transition from full numerical outputs of ensembles to in situ-generated ensembles necessitates extracting information or “features” from images to serve as the base for the interactive visualization pipeline. Several techniques have been developed to extract features from images, including but not limited to: global feature extraction, local feature extraction, and deep learning techniques. Global feature extraction techniques extract global properties, such as texture values, histogram, entropy, contrast, etc., from the entire image. This makes them less sensitive to noise but more prone to cluttering and occlusion. Conversely, local feature extraction techniques recognize patterns or objects, such as shapes, edges, lines, etc., from different regions of an image despite the clutter and occlusion. However, local features are hindered by the high dimensionality of the produced feature descriptors, which describe the image's distinctive characteristics (Zhen et al. 2017). The high dimensionality of the feature descriptors may result in high computational costs, noise, and redundant information, which could affect the accuracy of the visual analysis process.

Deep learning models have recently proven their effectiveness in producing extremely promising results in terms of image feature extraction. A key aspect of deep learning is its ability to learn and extract discriminative features from an image automatically. A deep learning model transforms low-level features of an image into high-level abstract features using an artificial neural network (Sun et al. 2018). However, adopting deep learning models to extract features from an image-based ensemble is highly challenging. Scientific data is a scarce resource, and each simulation model has its own characteristics. Therefore, obtaining sufficient labeled data for a supervised model would be hard. Even with an unsupervised model, there would not be enough training data, and the problem domain is narrow to have pre-trained models. This

could easily result in a low-accuracy model. Consequently, relying on a single feature extraction technique for extracting representative features from an image-based ensemble may not be sufficient for visual analysis, and exploration as the extracted features may have noise or unessential features.

This paper proposes a multi-level approach for feature extraction. First, a local feature extraction technique is used to extract feature descriptors for each image in the database. Then, a deep learning model is applied to extract key features from feature descriptors while discarding unnecessary ones. The reduced feature descriptors and simulation input parameters are then used in C-GLEE’s visualization pipeline.

3.2 System overview

Figure 1 provides the workflow of our approach, which consists of three primary components. First, given a high-dimensional simulation ensemble with different simulation input parameters, we employ in situ processing to visualize the output. This involves utilizing various visual mapping and view parameters to generate an image database. Thus, our approach begins with an ensemble $E = \{1, \dots, N\}$ of N members. Each ensemble member e_i ($i \in \{1, \dots, N\}$) consists of a set of input parameters (i.e., scalar fields) and an image database encoding simulation output from K different camera positions; $e_i = \{inputs, I_1, \dots, I_K\}$. We first apply scale-invariant feature transform (SIFT), a local feature extraction technique, to the image database members I_l ($l \in \{1, \dots, K\}$) for extracting significant features that preserve the intrinsic content of the images. Then, an autoencoder extracts important features from feature descriptors while eliminating superfluous features. Second, the reduced visual features and simulation input parameters are sent into similarity model, which, in turn, learns, manipulates, and projects the data into a 2D workspace space. Third, we build an interactive multi-view visual interface to simultaneously analyze and explore parameter and ensemble spaces.

C-GLEE’s multi-view visual interface (Fig. 2) and interaction techniques facilitate the analysis and exploration of image-based simulation ensembles. Ensemble members are visualized in C-GLEE’s ensemble view (Fig. 2a), which projects ensemble members in two-dimensional space via a projection technique (e.g., MDS, PCA, tSNE, etc.), using ensemble attributes (i.e., the simulation input parameters and reduced feature descriptors) and weights associated with them. The spatial arrangement of ensemble members in the ensemble view relies on both the attribute values and associated weights. C-GLEE positions ensemble members in low-dimensional space based on their similarity in high-dimensional space. Members with similar attributes are projected close to each other, while dissimilar ones are situated farther apart. Consequently, the initial projection of ensemble members or any projection resulting from an interaction may produce different layouts; the key aspect is that all these layouts maintain consistent distances between ensemble members in the low-dimensional space. Scientists can directly manipulate ensemble members through an observation-level interaction (OLI) technique, an interactive interaction technique designed based on the principles of semantic interaction. This interactive technique enables scientists to explore correlations and associations between and within the ensemble members using their intuition and expertise.

On the other hand, the input parameters are displayed as weights on a horizontal slider in the parameter view (Fig. 2b). The slider value represents the importance of that parameter within the similarity model. Scientists can explore parameter sensitivity by manipulating the value on the slider (i.e., increasing or decreasing) performing a parametric level interaction (PLI). PLI is used to understand and determine the sensitivity and influence of parameters on all ensemble members. Scientists might increase the weight of one or more attributes to learn which ensemble members are similar and different when focusing on these attributes. As additional attributes are up-weighted, scientists develop an understanding of how the attributes influence the data. Scientists can also use the Cinema slider (Fig. 2c) to navigate the image database, viewing the ensemble members from different phi–theta camera positions. A phi–theta camera is characterized by two angles, phi and theta, defining the view direction within a spherical coordinate system, where theta represents the polar angle while phi denotes the azimuthal angle. Manipulating the Cinema sliders (i.e., phi and theta controls) will result in an updated projection of ensemble members based on the visual features of images corresponding to the new view angle. These sliders enable the exploration and navigation of 3D data from different camera orientations (such as top, bottom, etc.). Cinema sliders thus offer scientists several perspectives into their ensemble results through the Cinema image database.

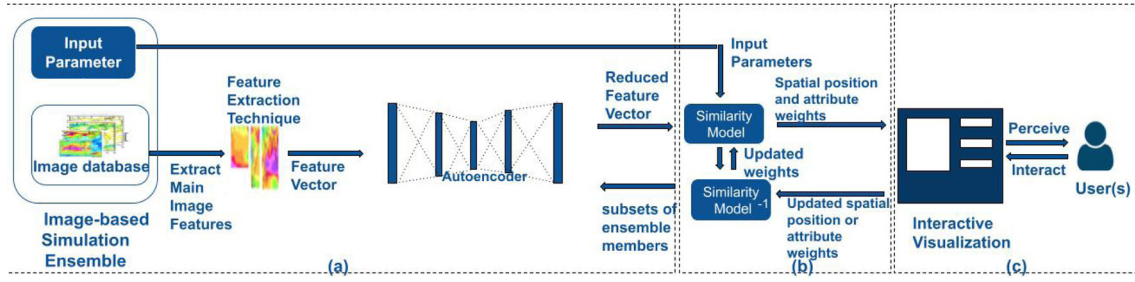


Fig. 1 Workflow of our approach. The workflow consists of three primary components: **a** the input source (attributes): the input parameters and reduced visual features from the image database. **b** The similarity model couples forward and backward models by translating semantic and parametric interactions into manipulations of model parameters that result in a new visualization. **c** An interactive multi-view visual interface that allows the exploration and analysis of the simulation ensemble



Fig. 2 C-GLUE's multi-view visual interface includes: **a** *Ensemble View* displays image thumbnails in a low-dimensional (2D) workspace representing the similarity and difference among ensemble members in the high-dimensional (parameter) space. Using weighted multi-dimensional scaling (WMDS), the image thumbnails are spatially organized in 2D space for interactive sorting by the scientist, **b** *Parameter View* shows the weights of simulation input parameters; the user can adjust the weights to explore the influence of parameters on the ensemble, and **c** *Cinema View* allows the selection of a certain viewing angles as thumbnails (from the Cinema image database)

4 C-GLUE image-based ensemble attributes

The in situ processing of the high-dimensional simulation ensemble involves employing different simulation input parameters, leading to the creation of an image database through the application of various visual mapping and view parameters. The simulation input parameters and image database serve as the entry point for our approach. However, before their integration into the C-GLUE's similarity model and visualization pipeline, a preprocessing step is implemented on all images in the databases. This entails extracting key features, starting with the utilization of SIFT local feature extraction, followed by the application of an autoencoder to reduce the dimensionality of the features extracted from SIFT. Subsequently, the input parameters and reduced feature descriptors are fed into the C-GLUE similarity model and visualization pipeline for visualization, exploration, and analysis.

4.1 Feature extraction

Given an ensemble of N members. Each ensemble member has an image database of K from different camera positions; $K = \theta * \phi$, where θ and ϕ represent theta and phi, respectively. All images in the image

databases are preprocessed by extracting key features that are then passed as input to C-GLEE. To start, SIFT is applied to every image I_l in the image database ($l \in \{1, \dots, K\}$). SIFT extracts distinctive invariant visual features from noisy, occluded, and cluttered images due to its high invariance to scaling, translation, and rotation. Specifically, SIFT detects potential pixels of interest in an image using a Difference-of-Gaussian (DoG) detector. Let $I_l(i, j)$ represents one pixel in image l , where i and j are pixel coordinates. The DoG detector starts by convolving all pixels in images with a Gaussian function $G(i, j, \sigma)$ (1) to create Gaussian-filtered images (F_l),

$$F_l(i, j, \sigma) = G(i, j, \sigma) * I_l(i, j), \quad (1)$$

where σ is a scaling argument of function $G()$. Then, a DoG image D_l is computed from the difference between baseline and scaled Gaussian-filtered images (2), where the scaled version inflates σ by g ,

$$D_l(i, j, \sigma) = F_l(i, j, g\sigma) - F_l(i, j, \sigma). \quad (2)$$

Next, SIFT selects feature key points from the local minimum or maximum of the Difference-of-Gaussians (DoG) smoothed images at different scales. After selection, each key point is assigned a principal orientation based on the local image-gradient direction, where the gradient magnitude (m) and orientation (θ) per image and pixel (i, j) are computed as follows:

$$m_l(i, j, \sigma) = \left((F_l(i+1, j, \sigma) - F_l(i-1, j, \sigma))^2 + (F_l(i, j+1, \sigma) - F_l(i, j-1, \sigma))^2 \right)^{1/2} \quad (3)$$

$$\theta_l(i, j, \sigma) = \tan^{-1} \left(\frac{F_l(i, j+1, \sigma) - F_l(i, j-1, \sigma)}{F_l(i+1, j, \sigma) - F_l(i-1, j, \sigma)} \right) \quad (4)$$

Finally, each key point and its neighboring pixels are transferred into feature descriptors. The feature descriptor of a key point is a vector of length 128 representing a 4x4 histogram array with eight orientation bins per histogram. That is, if P represents the number of key points plus nearest neighbors found in image $I_l(i, j)$, then SIFT produces a $P \times 128$ feature matrix.

Each image I_l is represented by a set of P key points, with 128 descriptors (128 dimensions) assigned to each key point. Typically, an image with dimensions of 512×512 pixels will generate around 2000 key points in total. As a result, the total number of key descriptors for such an image would be 2000×128 descriptors. Since the number of SIFT-detected key points varies among images within the database, some feature matrices are zero-padded. This ensures that all images in the database possess the same number of feature descriptors. The challenge with these feature descriptor vectors is their high dimensionality. To mitigate the curse of dimensionality, autoencoders are employed.

An autoencoder is an unsupervised artificial neural network model that learns meaningful features from high-dimensional image descriptors and transforms them into a low-dimensional representation while preserving the intrinsic structure of the data. An autoencoder is composed of two components: an encoder (C) and a decoder (D). The encoder translates the input data to a desired lossy compressed representation, known as a code layer or latent space representation, while the decoder decodes this code layer to an approximation of the inputs. The decoder is discarded after the training, and we are left with the encoder, which compresses the input data, reducing the dimensionality in a way that maximizes the salient features preserved in the input data.

The autoencoder reduces the dimensionality of the 128-dimensional SIFT feature descriptor vector for each key point to a Q -dimensional vector. The value of Q is dynamically chosen based on the characteristics of the image database. In this paper, Q was set to 2 or 5 based on the number of key points generated. Increasing Q above this threshold will result in a substantial increase in the number of visual feature descriptors. Such a combinatorial explosion of features has a direct impact on the number of calculations needed for the backward similarity model; this computational overhead is a challenge for C-GLEE to work in the real time regime.

In turn, the autoencoder is applied to transform the feature matrix from $P \times 128$ to $P \times Q$. The autoencoder architecture comprises three hidden layers, with a code layer containing Q nodes. This design is symmetric, maintaining an equal number of layers and nodes per layer in both the encoder and decoder. Consequently, the autoencoder is structured with a total of five layers, encompassing both input and output layers. The node configuration across the autoencoder architecture is as follows: (128 -> 16 -> Q -> 16 -

>128). The activation function applied to the encoder neurons is the sigmoid activation function, whereas the decoder neurons use the rectified linear unit (ReLU) activation function. The Adam optimization algorithm was used for stochastic optimization, along with mean squared error to compute losses.

The simulation input parameters and reduced feature descriptors are then supplied to C-GLEE similarity model and visualization pipeline for visual analysis and exploration of the ensemble.

5 C-GLEE image-based ensemble similarity model

C-GLEE similarity model leverages both input parameters and reduced feature descriptors (i.e., vectorized $P \times Q$ feature matrix) as the fundamental foundation for mapping data into the visualization outputs. Initially, to avoid misrepresentation during the visualization of the high-dimensional image-based ensemble, all the values are z-score normalized. Additionally, an initial weight vector is applied where a weight of $(1/A)$ is initially assigned to each ensemble's parameters and the visual features representing the image, where A represents the number of input parameters to the simulation and the reduced visual features derived from the image.

5.1 Similarity model

The similarity model comprises the forward and backward models, which translate user interactions into changes in the model's parameters. This leads to the generation of a new visualization.

The forward similarity model projects the high-dimensional ensemble into 2D space. C-GLEE could be programmed with any projection technique. However, we choosed to use weighted multi-dimensional scaling (WMDS) due to its facility in interpreting weighted dimensions, particularly in parameterizing user interactions. WMDS has shown to be easy to use and interpretable for both experts and non-experts. The input data to WMDS are normalized simulation input parameters and reduced feature descriptors for each image in the image database. WMDS combines a weighted distance function with multi-dimensional scaling to generate weighted projection of the image-based ensemble. The weighted distance function is used to capture explainable relationships (i.e., differences and similarities) among the ensemble members; explainable by input parameters and the image's visual features. The weighted distance function is chosen for an application is based on the nature of the task and the data. By default, C-GLEE uses weighted Euclidean distance function (5) is used; however, scientists can select other distance functions from C-GLEE's main interface, such as weighted Cosine, weighted Mahalanobis, weighted structural similarity index. To explore ensemble members e described by attributes based on both standardized simulation input parameters and extracted features from autoencoder, we apply WMDS using weighted Euclidean distance $D_w(e_i, e_j)$,

$$D_w(e_i, e_j) = \sqrt{\sum_{a=1}^A w_a (e_{i,a} - e_{j,a})^2}, \quad (5)$$

for ensemble members i and n ($i, j \in \{1, \dots, N\}$) with weight w_a applied to each attribute (i.e., feature or input) denoting its importance in the projections. The result of the pairwise Euclidean distances between ensemble members is then passed to WMDS. WMDS tries to find the low-dimensional locations of ensemble members by minimizing the mean squared error between the 2D and high-dimensional pairwise distance.

Conversely, the backward similarity model is invoked through the scientist's interaction with C-GLEE's visual interface using an OLI, a PLI, or the Cinema sliders. An OLI is a human-AI interaction technique that supports scientists in creating a tailored spatialization of their high-dimensional data based on their expertise and intuition. For example, based on the scientist's domain knowledge and expertise, s/he may disagree with the spatialization of the projected data or may visually observe an interesting pattern among some ensemble members. Therefore, s/he begins directly manipulating the subset of interested ensemble members, forming a cluster in an attempt to understand which attributes (i.e., those with high weights) may explain this cluster performing an OLI. This formed cluster expresses the scientist's hypothesized similarity between these ensemble members. C-GLEE learns the weights using a semi-supervised metric learning backward model. This model learns the new weights using new low-dimensional positions of the moved ensemble members, along with their corresponding high-dimensional points. The model starts by setting the weights for all

attributes to a uniform value. Subsequently, an optimization algorithm iteratively adjusts these weights until it converges, aiming to obtain a set of weights that reflect the updated positioning of the moved ensemble members. As a result, OLI empowers scientists to explore the correlations and relationships among and between ensemble members.

Scientists can directly explore and understand the influence of input parameters on simulation output using PLI. PLI is a user interaction that allows scientists to directly manipulate an attribute's weight on the slider. This manipulation results in an updated projection and weight vector. This allows scientists to provide parametric feedback to the similarity model regarding whatever attribute(s) s/he feels to be significant and to explore how this attribute(s) influences the relationships between different ensemble members. Scientists can also explore the features and properties of the ensemble using the Cinema slider to drive different visualization viewpoints (images) in the thumbnails in the OLI workspace. Manipulating the slider thus results in a new camera viewpoint for each ensemble member, which results in an updated projection of the ensemble based on the feature descriptors of the new view angle. By default, adjusting the Cinema slider resets the weight vector, and input parameters and image features are assigned the same weight. However, scientists have the freedom to either set the weight of attributes or maintain the old weights from the old viewpoint.

From one interaction to another, C-GLEE has no algorithmic memory. That is, weights learned from the optimization algorithm of the backward model after one OLI will not impact the optimization algorithm when invoked by another OLI or PLI. However, scientists accumulate knowledge over time. They might explore specific features in the data based on insights gained from previous interactions; however, there is no way of quantifying interactions order influence. As an exploration tool, C-GLEE does not endorse a singular "correct choice" of interaction or view angle. Scientists have the entire space of images at their disposal for exploration, enabling them to navigate and analyze the image dataset according to their specific objectives and hypotheses. C-GLEE's design encourages an open-ended exploration process, allowing scientists to iteratively investigate different angles, attributes, and projections to gain insights and make discoveries within the dataset.

6 C-GLEE image-based ensemble visualization and exploration

To implement the proposed approach as a visual analytical tool, we build an interactive multi-view visual interface. Figure 2 shows C-GLEE visual interface, including an *ensemble view*, a *parameter view*, and the *Cinema sliders*.

6.1 Multi-view visual interface

The ensemble view spatially arranges the low-dimensional projection of the image-based ensemble produced by the forward similarity model. Each ensemble member is represented by a 2D image of the simulation output. The ensemble view supports several interactions to explore the ensemble: OLI, multiple selection, lasso selection, and zooming. OLI enables scientists to directly manipulate the interesting subset of ensemble members based on their hypothesized similarity. This visual feedback is then translated into information that is fed as input to the backend similarity model, leading to updates in the weight vector and the members' projection accordingly.

C-GLEE's zooming functionality enables scientists to get finer details of thumbnail(s) of interesting ensemble member(s) for further exploration. Scientists could be interested in exploring a subset of ensemble members; therefore, C-GLEE supports multiple selection mechanisms, both lasso and multiple selections. These selection mechanisms allow scientists to alternate between two modes of analysis: overview first and details on demand.

The parameter view displays the weights of simulation input parameters on a horizontal slider. The slider's weights denote the significance of input parameters within the similarity model. Scientists manipulate the slider by increasing or decreasing its value(s) through PLI. This leads to an updated projection utilizing the updated weight vector, allowing scientists to explore and analyze parameter sensitivity. On the other hand, scientists adjust the Cinema sliders to explore significant features in an ensemble image database. This facilitates the comparison and contrast of different ensemble members from varied viewpoints, opening the door to new discoveries by exploring results through these viewpoints.

OLI results in an updated projection in the ensemble view and new weight values on sliders in the parameter view based on the learned similarities between clusters of ensemble members. PLI in the parameter view results in updating attribute weights on the slider and an updated projection in the ensemble view. The Cinema sliders are linked to the ensemble view; any adjustment in slider values results in updating the projection based on the image features of the new viewpoint.

7 Case studies

7.1 Simulation ensembles

We evaluated C-GLEE to assess its effectiveness using two ensemble simulations. The first simulation ensemble is a 2D dataset from oilfield wastewater disposal. The second dataset is a 3D ensemble of CO₂ sequestration. Our selection of datasets from the geosciences domain aims to examine C-GLEE's capability to manage scientific ensemble data that could run at the exascale. During the evaluation, we investigated how well the proposed workflow and C-GLEE's interaction techniques (OLI, PLI, and Cinema sliders) supported the scientists in exploring and analyzing their image-based ensemble results. The emphasis was on evaluating C-GLEE's effectiveness in capturing the intricate structure of the data contained in image databases and input parameters. C-GLEE functions as an exploratory visual analytical tool, enabling scientists to utilize various interactions that enhance the exploration process. The scientist has the flexibility to initiate the exploration with any combination of OLI interactions and PLI interactions. OLI and PLI represent two independent interaction techniques. Scientists have the flexibility to use them in any sequence during data exploration, depending on their personal expertise, needs, and curiosities. For instance, when scientists form hypotheses about relationships among ensemble members, they might employ OLI to identify attributes that could explain these relationships. On the other hand, when scientists assume correlations among two or more input parameters, they may use PLI to emphasize the importance of these attributes and assess how well the data aligns with these assumptions. However, with each new projection, scientists may choose to stop and summarize what they have learned, reapply the same interaction, or try a different interaction, depending on the insights gained.

7.1.1 Oilfield wastewater disposal

Our first case study evaluates the role of geologic and fluid properties on fluid pressure transients that trigger earthquakes during oilfield wastewater disposal (Pollyea et al. 2018). Recent research shows that fluid density plays an important role in fluid pressure build-up and earthquake triggering when oilfield wastewater is pumped into deep geologic formations via injection wells (Pollyea et al. 2019). Nevertheless, there remains substantial uncertainty in the correlation between fluid pressure build-up and 1) injection fluid temperature, 2) basement rock compressibility, and 3) basement rock permeability. The ensemble comprises numerical models of oilfield wastewater disposal using geologic and operational features of the Anadarko Shelf in southern Kansas and northern Oklahoma. Fifty ensemble members are reproduced from the same injection scenario and model domain, but they differ in the combination of spatially homogeneous injection fluid temperature, basement compressibility, and basement permeability. The simulation ensemble studied the influence of measurable parameters on the migration depth of fluid pressure after 10 years of wastewater disposal operations.

7.1.2 CO₂ saturation

The second case study focused on the effects of fracture-controlled reservoir heterogeneity in low-volume basalt reservoirs during geologic CO₂ sequestration (Pollyea et al. 2014). The permeability distribution was developed using sequential indicator simulation methods with constraints based on outcrop-scale fracture correlation models. The ensemble includes 3D synthetic reservoirs of 50 equally probable comprising a spatial permeability structure. It reflects basalt flow morphology in which the permeability of densely fractured flow tops/bottoms is $5\times$ greater than the lower permeability flow interiors.

7.2 Qualitative and quantitative analysis

To demonstrate the effectiveness of our approach, we compare the insights and conclusions generated by C-GLEE against the ground-truth conclusions derived during the scientist's regular analysis process. Typically, scientists employ visualizations of simulation outputs and summary statistics displays to study the variation of a single parameter (i.e., simulation outputs or input parameters) across all runs. This requires the use of multiple scripts or/and programs to visualize the data, which is usually time-consuming and error-prone. To facilitate post hoc exploration of in situ ensembles, we used the original input parameters and a Cinema dataset. Our challenge is then to extend the visualization and interaction pipelines to use the image database rather than numerical simulation outputs to answer the scientists' main questions. We used C-GLEE as the platform to test our method. We took special note of additional insights and conclusions that were not found during their traditional analysis.

We evaluated C-GLEE's image-based approach with three Geoscience domain scientists (a faculty member and two graduate students). The faculty member provided the ensemble data used in the experiments. We provided our tool to the domain experts to study relationships and similarities between ensemble members and parameter sensitivity. Initially, the scientists were given training on how to use C-GLEE and its interaction techniques. Later, we asked them to explore the ensemble using C-GLEE's multi-linked views and interaction techniques. During the evaluation, we measured the time taken to complete each task and the completeness of the task. We also measure the time taken by each performed interaction (i.e., OLI and PLI).

7.2.1 Expert evaluation—oilfield wastewater

The oilfield wastewater disposal simulation ensemble implements multi-physics numerical simulation methods that investigate fluid pressure transients that cause earthquakes. During the analysis of this ensemble, the scientist was trying to understand the correlation between fluid pressure build-up and rock permeability, injection fluid temperature, and rock compressibility. Therefore, s/he was interested in exploring if: (1) There are similarities between different ensemble members in which fluid pressure migrates deeper into the formation; and (2) if injection fluid temperature has any influence on fluid pressure build-up.

Figure 3 shows the initial projection of ensemble members (i.e., simulation outputs) in ensemble view. From these projections, the scientist observed that the ensemble members are arranged so that high and shallow magnitude pressure build-up are clustered on the one side, while lower and deeper magnitude pressure build-up are clustered on the other. The arrangement of ensemble members shows a more informative grouping of magnitude pressure build-up. To understand which parameter(s) governs this relationship, the scientist performs an OLI by grouping ensemble members into three clusters based on the migration depth of fluid pressure in the image representing each ensemble member, then clicking "Update" (Fig. 4). OLI is used to understand the similarities and differences between these ensemble members. The updated projection produced from this OLI led the scientist to gain an insight that permeability is the prominent parameter that governs the migration of fluid pressure depth (Fig. 5). Additionally, the clustering of ensemble members captured different behaviors (i.e., deep density-driven plumes and fluid pressure-driven down plumes). The insight concluded from using OLI allowed scientists to determine which parameter governs the similarities between different groups of ensemble members.

To explore if injection fluid temperature has any role on fluid pressure, the scientist manipulated the slider, increasing its value to make its importance, performing a PLI (Fig. 6). Based on the re-projection, the scientist concluded that ensemble members are arranged such that fluid pressure magnitude grows from top to bottom and fluid pressure depth increases from left to right. This leads the scientist to learn that the temperature of the fluid injection also affects the magnitude of fluid pressure buildup. Using PLI, the scientist was able to determine the influence and sensitivity of temperature on the ensemble. Additionally, the clustering of ensemble members in ensemble view accurately captures the second-order control on the depth of fluid migration.

The scientist continued using C-GLEE to explore the ensemble data. At the end, s/he concluded that (1) permeability governs the depth of pressure build-up, (2) for a given pressure depth, the changes in pressure magnitude are controlled by injection fluid temperature, and (3) rock compressibility does not impose any parametric control on depth or magnitude of fluid pressure transients caused by oilfield wastewater disposal.

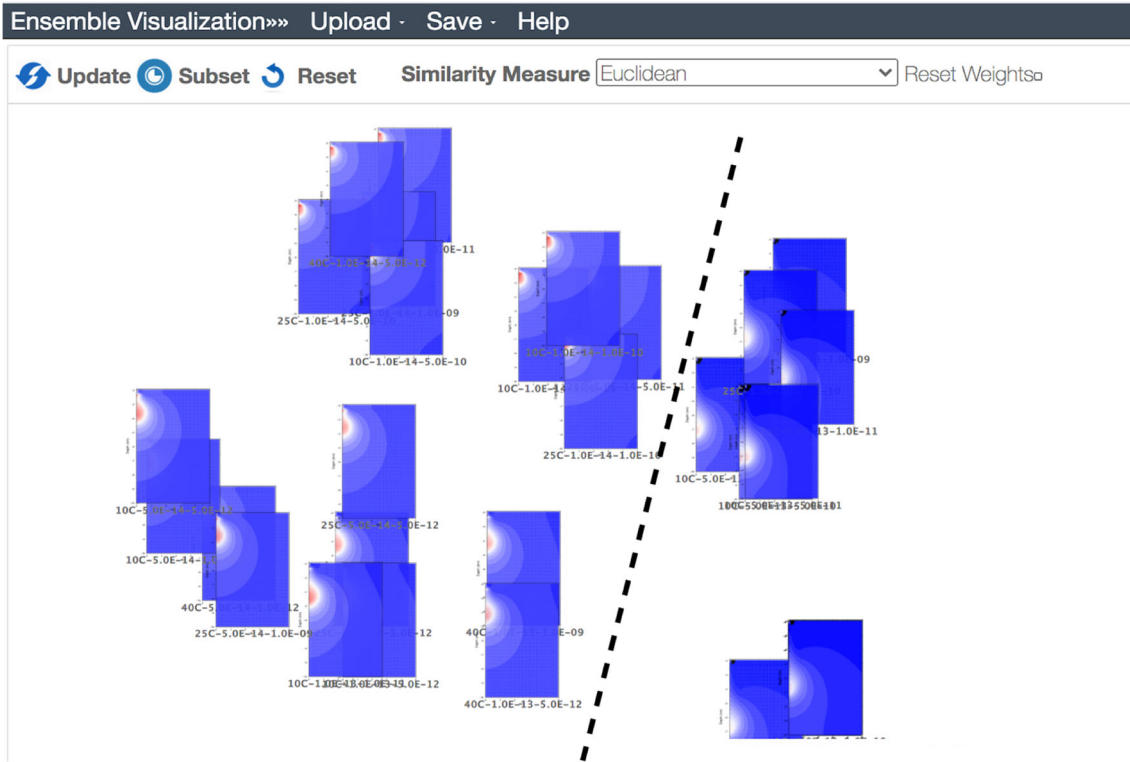


Fig. 3 The semantically sorted results from Oilfield wastewater disposal ensemble initial projection. The arrangement of ensemble members sorting shows that ensemble members are arranged so that high and shallow magnitude pressure build-up are clustered on one side, while lower and deeper magnitude pressure build-up are clustered on the other

7.2.2 Expert evaluation: CO_2 saturation

The CO_2 saturation simulation ensemble is focused on studying the structure of vertical CO_2 flow, which is governed by the spatial distribution of permeability. The initial projection appeared to cluster similarly shaped CO_2 plumes in close proximity, Fig. 7. The grouping of ensemble members captures this similarity more accurately. The scientist was interested in knowing how likely CO_2 escape is in the reservoir to find out if a CO_2 plume is leaking or not. Therefore, the scientist groups images with vertical CO_2 features (i.e., chimney-like features) based on their shape (wide fat or short tall). From the re-projection, the scientists observed that all CO_2 ensemble members with strong vertical expression are placed near the center, while the CO_2 plumes with more lateral continuity were projected in a ring around the center as shown in Fig. 8.

The scientists investigated the semantic impact of Cinema's different view angles on the ensemble by manipulating the Cinema slider. S/he manipulated the Cinema slider by changing to a view angle (i.e., $\theta = -75$ and $\phi = -30$) in the bottom of the reservoir colored by temperature, Fig. 9. This interaction resulted in an updated projection based on the image features of the view angle. This angle enables the scientist to explore the thermodynamics of the whole system, where the temperature cools in the middle of the reservoir. This is because the pushed CO_2 into the reservoir expands and cools down. The scientist noticed from the initial projection of this angle that the reservoir itself has an influence on the temperature regime of CO_2 , and there is some kind of control as part of the reservoir that needs further exploration. S/he then performs an OLI by grouping together ensemble members showing cooling, then clicking update Fig. 10. The resultant projection leads the scientist to observe that there are features worth exploring in more detail, requiring higher fidelity simulations near the borehole where the temperature could be used as a monitoring tool. Deriving new hypotheses such as this (for future testing) highlights the value of C-GLUE's multiple views into ensemble and its Cinema image database.

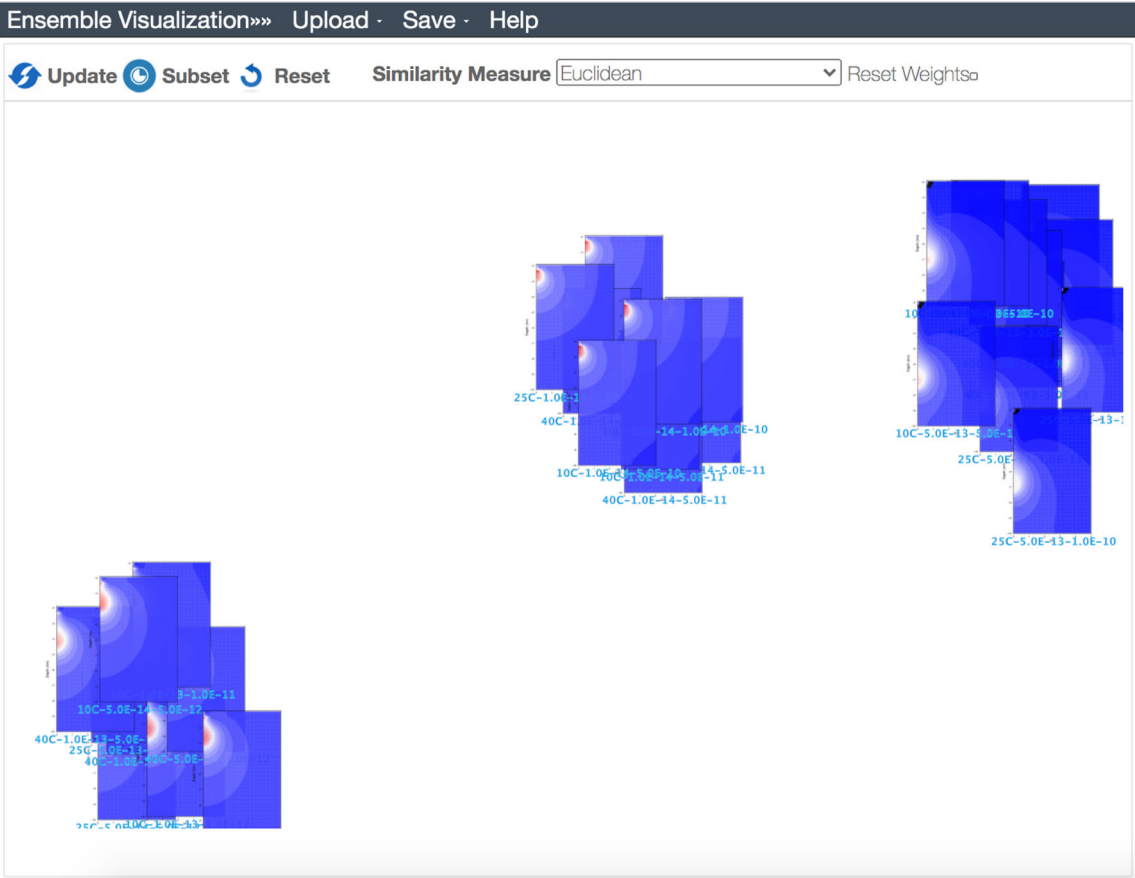


Fig. 4 Investigating migration of fluid pressure through OLI. Ensemble members are grouped into clusters based on fluid pressure depth from shallow (left) to deep (right), then “Update” button is clicked to get insights about this OLI interaction

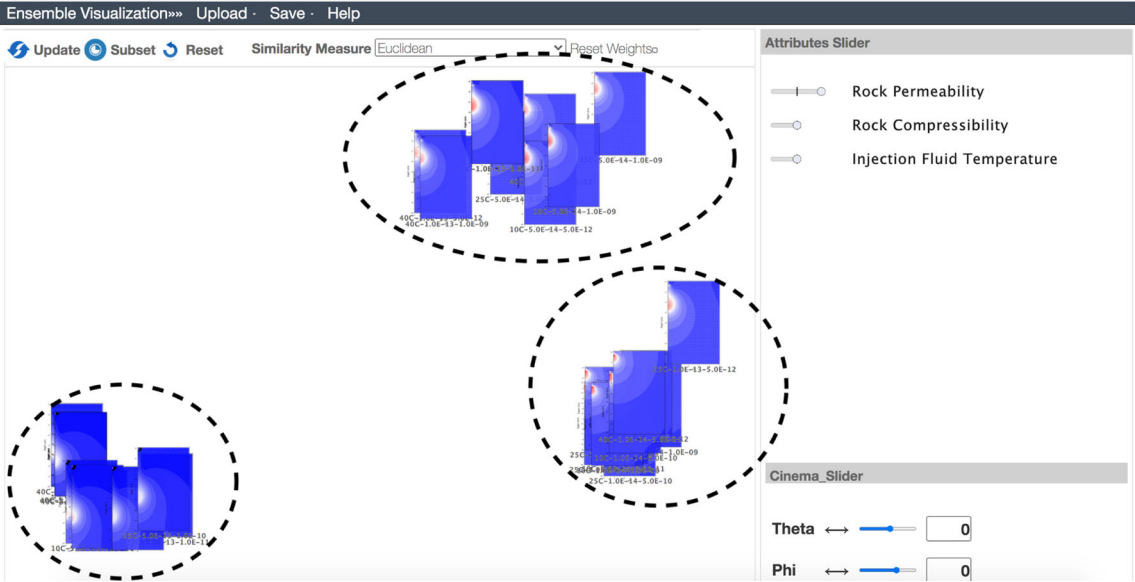


Fig. 5 The result of the OLI for ensemble members. From the re-projection, the scientist concluded that rock permeability controls the migration of fluid pressure depth

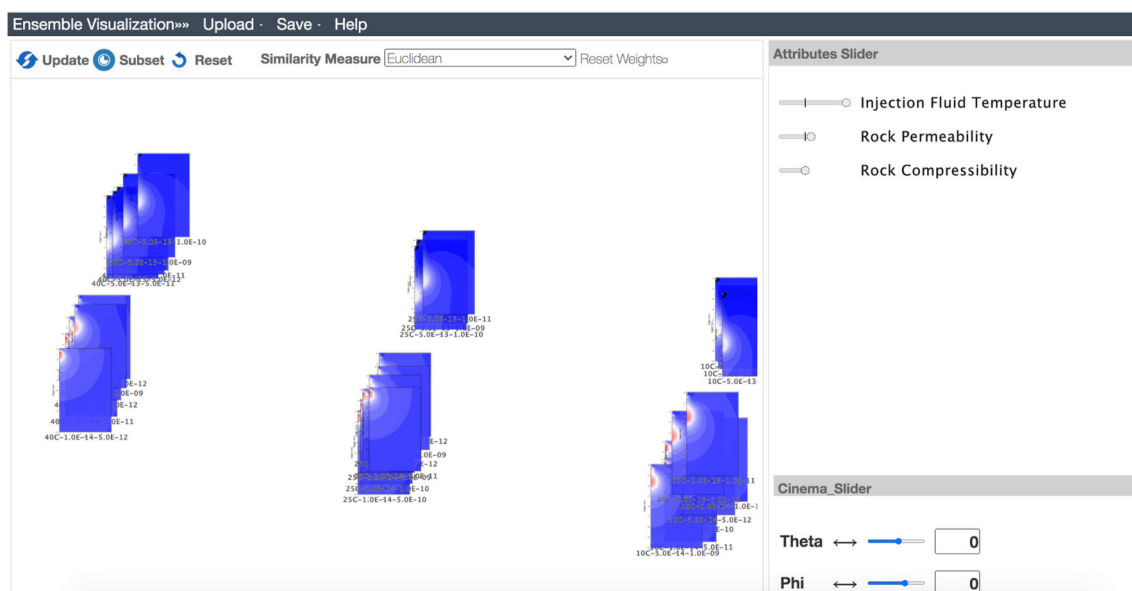


Fig. 6 The updated projection clusters ensemble members such that the magnitude of fluid pressure increases from top to bottom and the depth of fluid pressure grows from left to right. The scientist thus concluded that the injection temperature has an effect on the magnitude of fluid pressure

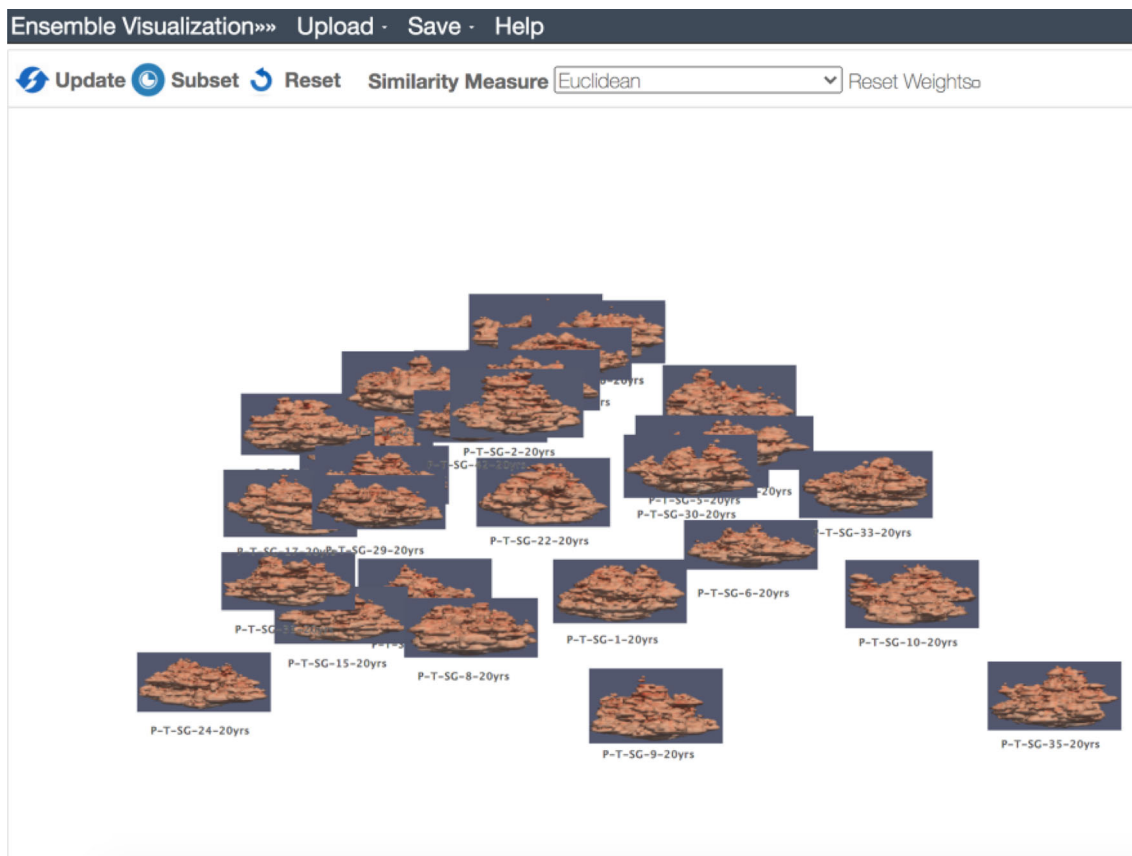


Fig. 7 The semantically sorted results from CO₂ sequestration ensemble. The initial projection appeared to cluster similarly shaped CO₂ plumes in close proximity

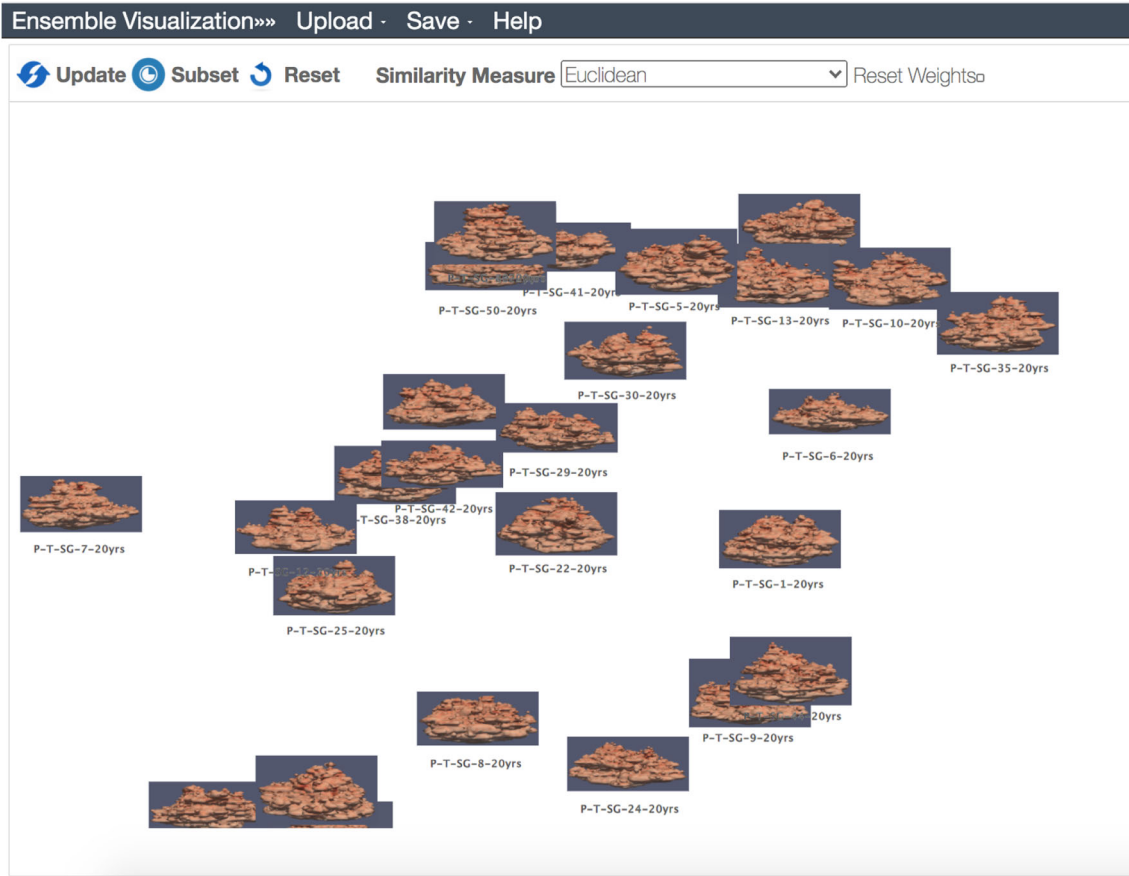


Fig. 8 The results of the OLI for grouping the vertical CO_2 features by geometry shape. From the re-projection, the scientist noticed that ensemble members with strong vertical CO_2 expression are placed near the center, while the others are projected in a ring around the center

8 Results

Cinema-GLEE (C-GLEE) is an exploratory visual analytics tool that helps scientists gain deeper insight into simulation ensembles when they lack a comprehensive understanding of the simulated model. It offers a new way of analyzing individuals relative to other ensemble members. Instead of relying on the whole ensemble's average variance or standard deviation for analysis, scientists can directly interact with and explore the relationships among different ensemble members. C-GLEE offers a novel approach for exploring high-dimensional ensembles when raw output data are unavailable. This differs from GLEE, which explores summary statistics of both the input and output data of individual ensemble members. C-GLEE allows scientists to test and explore hypotheses and ideas about the simulation data not only for a single image-based ensemble but rather an image database, creating new opportunities to discover new phenomena that may not be visible from a single view of the data.

The results of the qualitative analysis of the studies show that the majority of performed OLI enables scientists to figure out the similarities and differences among and between subsets of ensemble members. As an exploration interaction technique, OLI would not always yield meaningful results. It would be able to produce meaningful insights or even discoveries if the grouped ensemble members have common or similar features in the high-dimensional space that a metric learning model could capture. The updated projection from OLI enables scientists to determine which parameter(s) control the clustering of ensemble members. In some other re-projections, the visual features captured the similarities between ensemble members more than input parameters. In this case, the scientists did not observe a significant change in the values of input parameters on the sliders. However, grouping ensemble members after the projection led to new insights that significantly furthered the exploration process.

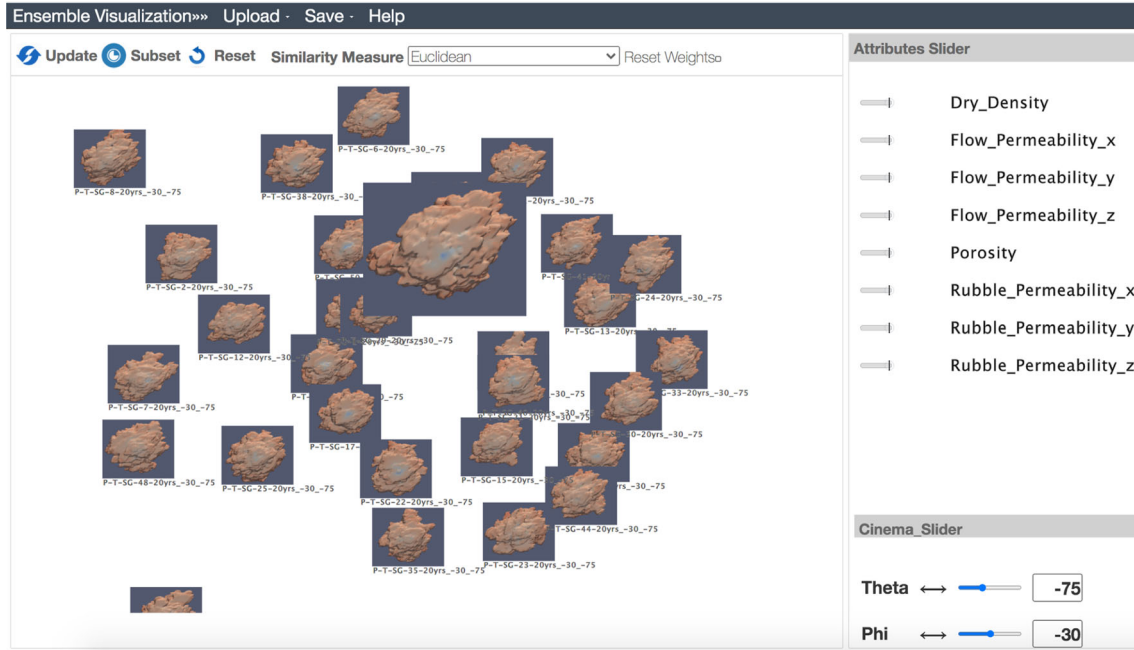


Fig. 9 The Cinema view at angle $\theta = -75$, $\phi = -30$ shows the bottom of the CO₂ saturation contour colored by temperature

Using PLI, scientists were able to determine the sensitivity of input parameters on visualized simulated output. They were able to find correlations between different input parameters. This helped scientists in determining important parameters and parameters that could be set to a constant. On the other hand, Cinema sliders offered them a chance to understand the correlations between ensemble members and parameter sensitivity across different viewing angles. This allowed them to compare and contrast the differences in correlation between ensembles from new perspectives. In some cases, the adjusted view angle did not produce clear insight, but it led to interesting observations or/and hypotheses that needed more high-fidelity simulation.

Our quantitative measurements indicate that C-GLEE's interaction techniques can empower scientists to complete all their initial exploration tasks. However, during interaction with C-GLEE, more exploratory questions were raised. While the scientists were able to find answers to some questions, others needed higher-fidelity simulations to answer them. We calculated the number of interactions taken to answer the initial exploration tasks. This number was different from one scientist to another depending on the type of interaction they performed first (OLI, PI, or Cinema) and the subset of ensemble members they explored. For example, the initial exploration tasks, on average, took 2–4 interactions to answer. More advanced tasks that were raised during the analysis vary in their number (an average of 4–7). Moreover, C-GLEE took a reasonable amount of time to respond to user interactions. It took less than 4 s, 2 s, and 1 s to respond to OLI, PLI, and Cinema sliders, respectively.

Looking across our methods and case studies, our proposed approach was able to reach the same conclusions that the scientists derived from their manual analysis process, but in significantly less time. Our approach allowed them to uncover new insights and findings that are hard to obtain with their traditional analysis process. Moreover, we observed that using SIFT and an autoencoder to extract image features preserved the intrinsic visual descriptors and was able to capture more complex structures in the data during the exploration. Moreover, the grouping of ensemble members in the ensemble view led to more insights derived by the scientists. Thus, the autoencoder can be extremely helpful when the dataset has a complex multi-dimensional structure. Recently, autoencoders have been used widely in several domains to extract important features from an image (Luo et al. 2019).

However, we believe that some limitations could hinder its direct adaptation to image-based simulation ensembles. For example, the storage and computational time required for training data. Training an image-based ensemble will need a large number of images to cover all the possible camera view angles and different simulation models; in case, the ensemble was constituted from different permutations of models. In this case, the size of the training data will be close to or greater than the raw simulation output data, which

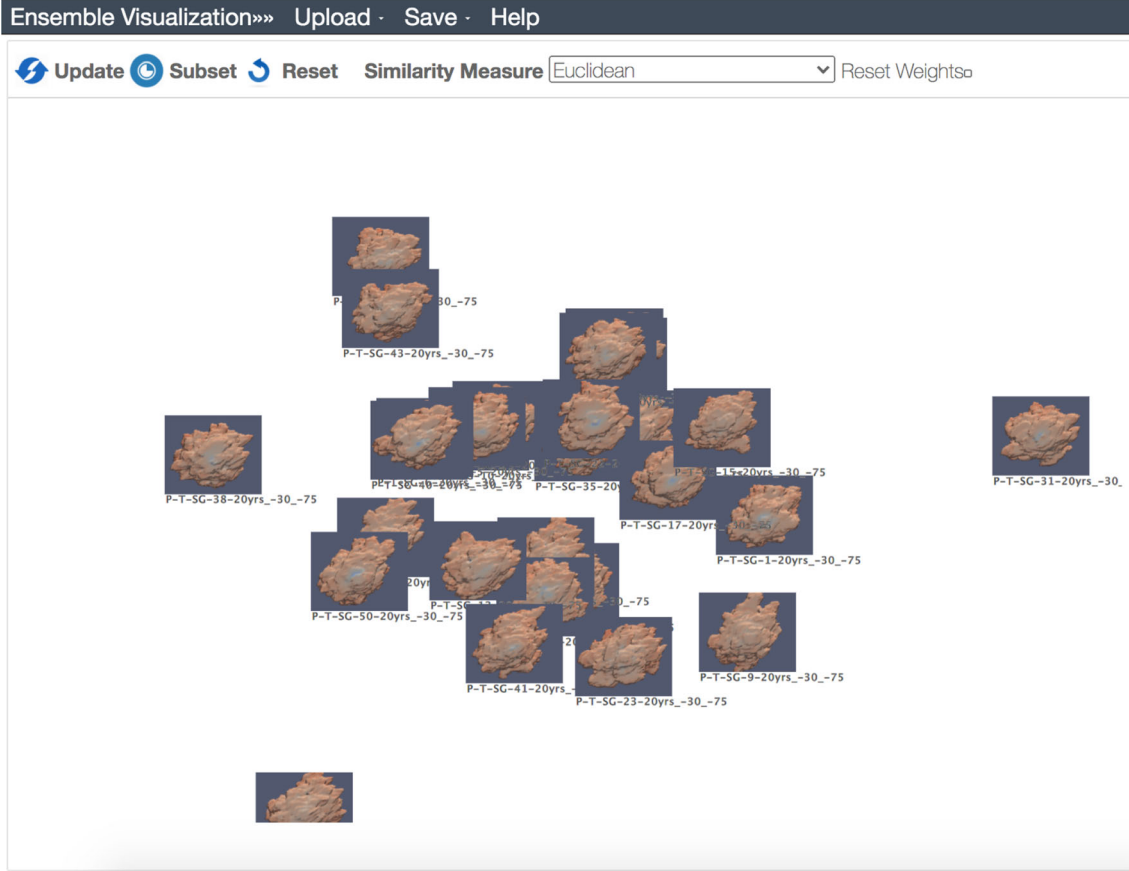


Fig. 10 The result of OLI interaction for the Cinema view at angle $\theta = -75$, $\phi = -30$. From the resulted projection, the scientist noticed that some specific features in the ensemble were worth exploring in more detail using higher-fidelity simulations

would counteract the advantages of using in situ visualization. Another limitation is cropping images or preprocessing steps to remove irrelevant or less important regions from the image (Xu et al. 2019). Moreover, passing image features produced directly by an autoencoder into C-GLEE’s visualization and interaction pipelines did not capture any complex structures in the data when OLI was used for exploration, unlike our currently proposed approach.

8.1 Limitations and future work

The main limitation of the proposed approach is scalability. Going to hundreds of ensemble members could easily lead to visual cluttering in the ensemble workspace. This problem could be partially solved by using larger displays that could accommodate more ensemble members or provide levels of detail summaries. Based on the feedback provided by scientists, we learned that they usually use an ensemble size that is smaller than one hundred. Moreover, C-GLEE’s backward similarity model performance would degrade in the case that the ensemble features and input parameters went over thousands of dimensions. For future work, we plan to explore other deep learning approaches. Cinema databases typically include time as a parameter and recent Cinema releases include non-image artifacts such as small meshes. Incorporating extended interaction techniques that include multiple views, time, or additional artifacts is another avenue for study.

9 Conclusion

In this paper, we explored the use of simulation ensembles visualized in situ within Cinema-GLEE (C-GLEE). To utilize exascale simulation ensemble outputs (i.e., Cinema image databases) as the input to C-GLEE, we needed to find an underlying statistical model to drive human expert–AI interaction (i.e., OLI and PLI). We achieved this by rephrasing GLEE’s visualization and interaction pipeline to handle this exascale data. We extracted feature descriptors from an image database using a local feature extraction technique and a deep learning model. In addition, an interactive visual interface was developed to explore commonalities and dissimilarities between ensemble members and determine parameter sensitivity. We illustrated the effectiveness of our proposed approach with several experiments with geoscience domain experts. We found that using C-GLEE, scientists could effectively explore the simulation space with their image databases and find new insights. This is important as simulations move to exascale and full numerical output data may not be available for post-processing and visualization. Also, by validating image-based techniques, a human expert–AI partnership could be applied to experimental image-based data, allowing scientists a different approach to finding connections in experimental data.

Acknowledgements This work was supported by the US Department of Energy through the Los Alamos National Laboratory. Los Alamos National Laboratory is operated by Triad National Security, LLC, for the National Nuclear Security Administration of the US Department of Energy (Contract No. 89233218CNA000001). This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the US Department of Energy Office of Science, and the National Nuclear Security Administration. This work was released under LA-UR-21-23424.

References

- Ahrens J, Jourdain S, O’Leary P, et al (2014) An image-based approach to extreme scale in situ visualization and analysis. In: Proceedings of the international conference for high performance computing, networking, storage and analysis. IEEE Press, Piscataway, NJ, USA, SC ’14, pp 424–434. <https://doi.org/10.1109/SC.2014.40>
- Bauer AC, Abbasi H, Ahrens J, et al (2016) In situ methods, infrastructures, and applications on high performance computing platforms. In: Computer Graphics Forum, Wiley Online Library, pp 577–597. <https://doi.org/10.1111/cgf.12930>
- Bensema K, Gosink L, Obermaier H et al (2015) Modality-driven classification and visualization of ensemble variance. IEEE Trans Visual Comput Graphics 22(10):2289–2299. <https://doi.org/10.1109/TVCG.2015.2507569>
- Chen X, Li J, Zhang Y et al (2020) Automatic feature extraction in x-ray image based on deep learning approach for determination of bone age. Futur Gener Comput Syst 110:795–801. <https://doi.org/10.1016/j.future.2019.10.032>
- Dahshan M, Polys N, Jayne R, et al (2020) Making sense of scientific simulation ensembles with semantic interaction. In: Computer Graphics Forum, Wiley Online Library, pp 325–343. <https://doi.org/10.1111/cgf.14029>
- de Souza CVF, Barcellos PCL, Crissaff L et al (2022) Visualizing simulation ensembles of extreme weather events. Comput Graph 104:162–172. <https://doi.org/10.1016/j.cag.2022.01.007>
- Di S, Cappello F (2016) Fast error-bounded lossy hpc data compression with sz. In: 2016 IEEE international parallel and distributed processing symposium (ipdps), IEEE, pp 730–739. <https://doi.org/10.1109/IPDPS.2016.11>
- Fernandes O, Frey S, Sadlo F, et al (2014) Space-time volumetric depth images for in-situ visualization. In: 2014 IEEE 4th symposium on large data analysis and visualization (LDAV), IEEE, pp 59–65. <https://doi.org/10.1109/LDAV.2014.7013205>
- Gao S, Duan L, Tsang IW (2015) Defeatnet: a deep convolutional image representation for image classification. IEEE Trans Circuits Syst Video Technol 26(3):494–505. <https://doi.org/10.1109/TCSVT.2015.2389413>
- He N, Paoletti ME, Haut JM et al (2018) Feature extraction with multiscale covariance maps for hyperspectral image classification. IEEE Trans Geosci Remote Sens 57(2):755–769. <https://doi.org/10.1109/TGRS.2018.2860464>
- He W, Wang J, Guo H et al (2019) Insitunet: deep image synthesis for parameter space exploration of ensemble simulations. IEEE Trans Visual Comput Graphics 26(1):23–33. <https://doi.org/10.1109/TVCG.2019.2934312>
- Höllt T, Magdy A, Zhan P et al (2014) Ovis: a framework for visual analysis of ocean forecast ensembles. IEEE Trans Visual Comput Graphics 20(8):1114–1126. <https://doi.org/10.1109/TVCG.2014.2307892>
- Hummel M, Obermaier H, Garth C et al (2013) Comparative visual analysis of Lagrangian transport in cfd ensembles. IEEE Trans Visual Comput Graphics 19(12):2743–2752. <https://doi.org/10.1109/TVCG.2013.141>
- Kovalchuk SV, Boukhanovsky A (2015) Towards ensemble simulation of complex systems. In: ICCS, pp 532–541. <https://doi.org/10.1016/j.procs.2015.05.280>
- Kumpf A, Rautenhaus M, Riemer M et al (2018) Visual analysis of the temporal evolution of ensemble forecast sensitivities. IEEE Trans Visual Comput Graphics 25(1):98–108. <https://doi.org/10.1109/TVCG.2018.2864901>
- Kumpf A, Stumpfegger J, Härtl PF et al (2021) Visual analysis of multi-parameter distributions across ensembles of 3d fields. IEEE Trans Visual Comput Graphics 28(10):3530–3545. <https://doi.org/10.1109/TVCG.2021.3061925>
- Kunang YN, Nurmaini S, Stiawan D, et al (2018) Automatic features extraction using autoencoder in intrusion detection system. In: 2018 International Conference on Electrical Engineering and Computer Science (ICECOS), IEEE, pp 219–224. <https://doi.org/10.1109/ICECOS.2018.8605181>
- Leistikow S, Nahardani A, Hoerr V, et al (2020) Interactive visual similarity analysis of measured and simulated multi-field tubular flow ensembles. In: Eurographics Workshop on Visual Computing for Biology and Medicine. The Eurographics Association, pp 139–150. <https://doi.org/10.2312/vcbm.20201180>

- Luciani T, Burks A, Sugiyama C et al (2018) Details-first, show context, overview last: supporting exploration of viscous fingers in large-scale ensemble simulations. *IEEE Trans Visual Comput Graphics* 25(1):1–11. <https://doi.org/10.1109/TVCG.2018.2864849>
- Lukasczyk J, Garth C, Larsen M, et al (2020) Cinema darkroom: a deferred rendering framework for large-scale datasets. In: 2020 IEEE 10th Symposium on Large Data Analysis and Visualization (LDAV), IEEE, pp 37–41. <https://doi.org/10.1109/LDAV51489.2020.00011>
- Luo X, Li X, Wang Z et al (2019) Discriminant autoencoder for feature extraction in fault diagnosis. *Chemom Intell Lab Syst* 192:103814. <https://doi.org/10.1016/j.chemolab.2019.103814>
- Ma B, Entezari A (2018) An interactive framework for visualization of weather forecast ensembles. *IEEE Trans Visual Comput Graphics* 25(1):1091–1101. <https://doi.org/10.1109/TVCG.2018.2864815>
- Maack RG, Rogers DH, Hagen H, et al (2020) Exploring cinema databases using multi-dimensional image measures
- Mahajan S, Gaddis AL, Evans KJ et al (2017) Exploring an ensemble-based approach to atmospheric climate modeling and testing at scale. *Procedia Computer Sci* 108:735–744. <https://doi.org/10.1016/j.procs.2017.05.259>
- Meyer M, Pfister H, Hansen C, et al (2005) Image-based volume rendering with opacity light fields. No UUSCI-2005-002 Tech Report
- Mirzargar M, Whitaker RT, Kirby RM (2014) Curve boxplot: generalization of boxplot for ensembles of curves. *IEEE Trans Visual Comput Graphics* 20(12):2654–2663. <https://doi.org/10.1109/TVCG.2014.2346455>
- Orban D, Keefe DF, Biswas A et al (2018) Drag and track: a direct manipulation interface for contextualizing data instances within a continuous parameter space. *IEEE Trans Visual Comput Graphics* 25(1):256–266. <https://doi.org/10.1109/TVCG.2018.2865051>
- Orban D, Banesh D, Tauxe C et al (2020) Cinema: Bandit: a visualization application for beamline science demonstrated on xfel shock physics experiments. *J Synchrotron Radiat* 27(1):1–10. <https://doi.org/10.1107/S1600577519014322>
- Petz C, Pöthkow K, Hege HC (2012) Probabilistic local features in uncertain vector fields with spatial correlation. *Computer Graph Forum* 31(3pt2):1045–1054. <https://doi.org/10.1111/j.1467-8659.2012.03097.x>
- Ping Tian D et al (2013) A review on image feature extraction and representation techniques. *Int J Multim Ubiquit Eng* 8(4):385–396
- Pollyea RM, Fairley JP, Podgorney RK et al (2014) Physical constraints on geologic CO₂ sequestration in low-volume basalt formations. *GSA Bull* 126(3–4):344–351. <https://doi.org/10.1130/B30874.1>
- Pollyea RM, Mohammadi N, Taylor JE et al (2018) Geospatial analysis of Oklahoma (USA) earthquakes (2011–2016): quantifying the limits of regional-scale earthquake mitigation measures. *Geology* 46(3):215–218. <https://doi.org/10.1130/G39945.1>
- Pollyea RM, Chapman MC, Jayne RS et al (2019) High density oilfield wastewater disposal causes deeper, stronger, and more persistent earthquakes. *Nat Commun*. <https://doi.org/10.1038/s41467-019-11029-8>
- Ribés A, Poudroux J, Iooss B (2019) A visual sensitivity analysis for parameter-augmented ensembles of curves. *J Verif Valid Uncertain Quantif*. <https://doi.org/10.1115/1.4046020>
- Risojević V, Babić Z (2012) Fusion of global and local descriptors for remote sensing image classification. *IEEE Geosci Remote Sens Lett* 10(4):836–840. <https://doi.org/10.1109/LGRS.2012.2225596>
- Sanyal J, Zhang S, Dyer J et al (2010) Noodles: a tool for visualization of numerical weather model ensemble uncertainty. *IEEE Trans Visual Comput Graphics* 16(6):1421–1430. <https://doi.org/10.1109/TVCG.2010.181>
- Sedlmair M, Heinzl C, Bruckner S et al (2014) Visual parameter space analysis: a conceptual framework. *IEEE Trans Visual Comput Graphics* 20(12):2161–2170. <https://doi.org/10.1109/TVCG.2014.2346321>
- Shareef N, Lee TY, Shen HW, et al (2006) An image-based modelling approach to gpu-based unstructured grid volume rendering. In: *Volume Graphics*, pp 31–38. <https://doi.org/10.2312/VG/VG06/031-038>
- Sun M, Konstantelos I, Strbac G (2018) A deep learning-based feature extraction framework for system security assessment. *IEEE Trans Smart Grid* 10(5):5007–5020. <https://doi.org/10.1109/TSG.2018.2873001>
- Tikhonova A, Correa CD, Ma KL (2010) Explorable images for visualizing volume data. *PacificVis* 10:177–184. <https://doi.org/10.1109/PACIFICVIS.2010.5429595>
- Torsney-Weir T, Saad A, Moller T et al (2011) Tuner: principled parameter finding for image segmentation algorithms using visual response surface exploration. *IEEE Trans Visual Comput Graphics* 17(12):1892–1901. <https://doi.org/10.1109/TVCG.2011.248>
- Tuyltaars T, Mikolajczyk K, et al (2008) Local invariant feature detectors: a survey. *Foundations and trends®. Comput Graph Vision*, 3(3):177–280. <https://doi.org/10.1561/06000000017>
- Wang J, Hazarika S, Li C et al (2018) Visualization and visual analysis of ensemble data: a survey. *IEEE Trans Visual Comput Graphics* 25(9):2853–2872. <https://doi.org/10.1109/TVCG.2018.2853721>
- Xu K, Xia M, Mu X et al (2018) EnsembleLens: ensemble-based visual exploration of anomaly detection algorithms with multidimensional data. *IEEE Trans Visual Comput Graphics* 25(1):109–119. <https://doi.org/10.1109/TVCG.2018.2864825>
- Xu W, Keshmiri S, Wang G (2019) Adversarially approximated autoencoder for image generation and manipulation. *IEEE Trans Multimedia* 21(9):2387–2396. <https://doi.org/10.1109/TMM.2019.2898777>
- Ye YC, Wang Y, Miller R, et al (2015) In situ depth maps based feature extraction and tracking. In: 2015 IEEE 5th symposium on large data analysis and visualization (LDAV), IEEE, pp 1–8. <https://doi.org/10.1109/LDAV.2015.7348065>
- Zhang Y, Li G, Yue R et al (2022) Peviz: an in situ progressive visual analytics system for ocean ensemble data. *J Visualiz.* <https://doi.org/10.1007/s12650-022-00883-2>
- Zhen X, Zheng F, Shao L et al (2017) Supervised local descriptor learning for human action recognition. *IEEE Trans Multimedia* 19(9):2056–2065. <https://doi.org/10.1109/TMM.2017.2700204>