This article has been accepted for publication in IEEE Transactions on Visualization and Computer Graphics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2022.3232367

Spatio-Temporal Visual Analysis of Turbulent Superstructures in Unsteady Flow

Behdad Ghaffari, Davide Gatti, and Rüdiger Westermann

Abstract—The large-scale motions in 3D turbulent channel flows, known as Turbulent Superstructures (TSS), play an essential role in the dynamics of small-scale structures within the turbulent boundary layer. However, as of today, there is no common agreement on the spatial and temporal relationships between these multiscale structures. We propose a novel space-time visualization technique for analyzing the temporal evolution of these multiscale structures in their spatial context and, thus, to further shed light on the conceptually different explanations of their dynamics. Since the temporal dynamics of TSS are believed to influence the structures in the turbulent boundary layer, we propose a combination of a 2D space-time velocity plot with an orthogonal 2D plot of projected 3D flow structures, which can interactively span the time and the space axis. Besides flow structures indicating the fluid motion, we propose showing the variations in derived fields as an additional source of explanation. The relationships between the structures in different spatial and temporal scales can be more effectively resolved by using various filtering operations and image registration algorithms. To reduce the information loss due to the non-injective nature of projection, spatial information is encoded into transparency or color. Since the proposed visualization is heavily demanding computational resources and memory bandwidth to stream unsteady flow fields and instantly compute derived 3D flow structures, the implementation exploits data compression, parallel computation capabilities, and high memory bandwidth on recent GPUs via the CUDA compute library.

Index Terms—Flow visualization, large-scale data techniques, animation and motion-related techniques

1 Introduction

The hallmark of turbulence in even the simplest wall flows, such as turbulent flow in a plane channel, is their three-dimensional, unsteady and multiscale behavior, making their study via statistical analysis or flow visualization challenging. In fact, vortical structures with different characteristic length scales coexist and interact with each other in wall-bounded turbulence. They range from small-scale eddies in the wall vicinity to very large-scale structures away from the wall. The latter are also called turbulent superstructures — TSS in the following – and have length scales much larger than those of the near-wall vortices [1], [2].

While there is an agreed-upon view of the dynamics, origin and evolution of near-wall eddies [3], the mechanism that yields the formation of TSS has not yet been fully understood and is an object of active research. TSS occur predominantly in high-Reynolds number turbulent flows, such as most technical and geophysical flows, whose properties are then strongly determined by the presence of TSS. TSS play an important role in the dynamics of wall turbulence [4], [5], where they are indicated by pronounced spatio-temporal streak-like patterns in the velocity fluctuation field at length scales of many times the largest characteristic length scale, L, of the flow, such as the height $2h$ for channels, where h is half the channel height which is also known as channel semi-height. In particular, TSS are known to have a modulating effect on the other eddies populating the turbulent flow [6], [7], [8], in particular on the smallest near-wall vortices, which are responsible for the self-sustainment of turbulence [3].

Understanding how TSS form and interact with smaller eddies in the turbulent flow is made particularly difficult by the requirement to simultaneously analyze different types of flow structures at different length- and time-scales. On the one hand, large-enough portions of the flow have to be observed for a long time, to encompass several TSS and their slow temporal evolution. Yet, the spatial and temporal resolution must be sufficient to resolve the size and dynamics of the smallest eddies. The combination of these aspects makes the visual inspection of TSS an open challenge, where the ultimate goal is to detect causal relationships between different types of dynamically-evolving flow structures at different scales. Since there are structures in turbulent flows which are not advected with the local flow field velocity [9], [10], such as velocity fluctuation at different scales, wall-shear stress or certain vortex indicators, alternative mechanisms to extract the motion dynamics of such structures are required.

In this work, we shed light on the mechanisms driving the temporal formation of multiscale structures by analyzing the relationships between spatio-temporal TSS in the velocity fluctuation field in 3D channel flows, so-called velocity streaks, and other emergent flow patterns as potential causes of their evolution. Such streaks can clearly be seen when visualizing the temporal evolution of the mean-removed streamwise velocity component, indicating tree-like merge and split events of streaks over time. In fluid dynamics, Eulerian space-time plots are often used to visualize this evolution (see Fig. 1). They show for a single

- *Behdad Ghaffari and Rüdiger Westermann are with Technical University of Munich, Germany. E-mail: behdad.ghaffari, westermann@tum.de.*
- *Davide Gatti is with Karlsruhe Institute of Technology, Karlsruhe, Germany. Email: davide.gatti@kit.edu.*

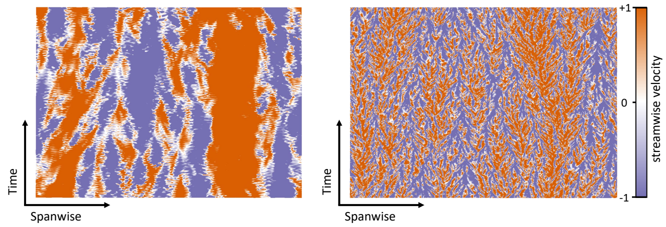

Fig. 1. Space-time plots of the velocity fluctuation field in a 3D turbulent channel flow. Positive and negative streamwise component of the velocity fluctuations, i.e., the streamwise component of mean-removed velocities, are shown in orange and blue, respectively. Left: Large-scale structures exist predominantly away from the boundaries at a wallnormal position $y = h$. Right: Small-scale structures reside in wall vicinity at $y = 0.05h$.

seed line the velocity magnitudes over time in a 2D plane.

Despite the clear evidence of the existence of the described TSS, as of today, there is no common agreement on the spatial and temporal relationships between these TSS and other flow structures. What makes the visual analysis of the mechanisms driving these TSS difficult is their temporal nature. Since a space-time plot uses one dimension to show the temporal evolution, visualizations of 3D flow structures cannot be directly combined due to the unavoidable misalignment of spatial and temporal dimensions. Furthermore, while it is possible to generate a space-time plot at rates that allow for an interactive re-positioning of the seed line, simultaneous access to other flow structures in an unsteady field becomes difficult. In particular, since domain experts want to flexibly probe different flow structures and maybe even combinations of them, any effort to pre-compute and store these structures becomes difficult due to memory constraints.

Contribution

We propose a novel space-time visualization technique and its efficient realization for analyzing the temporal evolution of TSS in their spatial context. By this, we aim to further shed light on the validity of the conceptually different explanations of the causal relationships between multiscale turbulent structures. To obtain the possibility to investigate the mutual spatial and temporal relationships between these patterns, we propose a combination of the commonly used 2D space-time velocity plot with an orthogonal 2D plot of projected 3D flow structures. The motivation underlying this specific design is to enable animating through the time span and simultaneously show the current TSS along the seed line and the emergent flow structures in spatial space. Since, in the projection, certain spatial relationships are lost, we provide visual cues using color and transparency to indicate spatial proximity.

The system provides the visualization of pathlines to show the fluid motion, as well as streamlines to indicate the instantaneous structure of the velocity fields and enable to explore the relation of turbulent superstructures to these structures. Using short streamlines, the tangential velocity in the projection plane is visualized in a glyph-like manner. The large-scale structures addressed in the present manuscript are classically defined as an Eulerian feature of the fluctuation field, typically as positive or negative isosurfaces of a filtered version of the streamwise fluctuation velocity. This definition results in surfaces, that are not material surfaces, i.e., not necessarily made up of the same material particle tracers. Since their definition is related to the instantaneous velocity fluctuation field, we also use streamlines as a means to visualize them. In addition, the system enables interactive analysis of so-called motion fields, which are derived from arbitrary quantities that are passively or actively advected by the flow. Motion fields are computed via the optical flow between simulated or derived scalar fields at two adjacent timesteps, and they are used as an additional source of explanation to investigate advection processes that are not driven by the local flow field velocity.

Since the proposed visualization heavily demands computational resources and memory bandwidth to stream unsteady flow fields and instantly compute derived 3D flow structures. Our implementation exploits lossy data compression, parallel computation capabilities, and high memory bandwidth on recent GPUs via the CUDA compute library. We have built into the visualization tool a lossy GPU compression layer that reduces the data to a volume that can be streamed efficiently so that on-the-fly computation of emergent flow structures is possible. We demonstrate that the errors introduced by a lossy compression scheme do not cause changes in the overall structure of the computed flow structures. In particular, since such changes are local and occur only sporadically, resulting errors do not hinder a phenomenological visual analysis as proposed.

Our specific contributions are:

- An enhanced 2.5D space-time plot that can show the temporal relationships between flow field velocity and other scalar flow quantities.
- An interactive space-time visualization technique combining 2.5D views of temporal velocity variations and projected 3D flow structures enhanced by proximity cues.
- The visualization of advection processes that are not explained by the flow velocity via the optical flow between derived feature fields at adjacent timesteps.
- A high-performance GPU visualization system that enables the efficient computation and visualization of particle-based flow structures and derived scalar feature fields in 3D time-dependent fields.

Besides the application of the proposed technique to turbulent 3D channel flow, it can be applied to many related time-dependent problems in computational fluid dynamics, such as the investigation of the origin of near-wall streaks [11]. In the following, after describing the specific dataset we analyze in this work, we introduce the elementary data processing operations that are applied to enable an interactive visual time-space analysis of this dataset. We further provide an overview of the workflow we have designed to address the domain-specific questions. Our novel extension of the 2D space-time plot and its combination with a visualization of projected time-varying 3D flow structures is discussed next. We conclude the paper with a discussion of implementation aspects and the current results and sketch future applications of the proposed visualization technique. The code is made publicly available under a BSD license and is published on https://github.com/ghaffaribehdad/Flow_visualization.

2 Related Work

Our work is mainly related to visualization techniques for analyzing the motion dynamics in turbulent flows, both with respect to the flow changes from timestep to timestep and transport structures in the instationary flow. On the one hand, our approach makes use of several established so-called particle-based flow visualization techniques [12], [13]. In particular, we utilize parallel computing capabilities and high memory bandwidth on Graphics Processing Units (GPUs) to trace large numbers of particles at interactive rates [14], [15], [16], [17]. When the data becomes too large to be stored on the GPU, computations are typically carried out on parallel compute clusters. Thorough overviews regarding the specific parallelization and memory layout strategies are provided in [18], [19], [20], [21]. Recent work by Guo et al. [22] discusses the different concepts underlying parallel particle tracing algorithms in the context of stochastic particle tracing for uncertainty analysis. This includes a discussion of different block partitioning and assignment strategies, as well as the use of parallelization over the set of particles and using on-demand fetching of required blocks. On a 32 Blue Gene/Q nodes cluster, they report roughly one second for the integration of 32K particles in the unsteady Visualization Contest 2004 data set of resolution $500 \text{x} 500 \text{x} 100 \text{x} 4$ (temporal resolution downsampled by a factor of 12). Our system computes this in about 400 milliseconds on a single node GPU, including streaming the compressed data from the CPU and decoding on the GPU.

A single-node system for large-scale turbulence visualization has been proposed by Treib et al. [23]. It addresses data scalability via a lossy data compression layer and data-parallel ray-based visualization on the GPU. However, this system does not support particle-based flow visualization but visualizes scalar features in turbulent flows derived locally from the velocity gradient tensor, i.e., using classical vortex criteria like λ_2 [24]. Bürger et al. [15] use the same compression layer and analyze the accumulation of reconstruction errors during particle integration, which can lead to trajectories that are different from those computed in the uncompressed field. They argue that the trajectories we see in the initial field are always afflicted with uncertainty due to interpolation and integration errors as well as approximations in the numerical simulation schemes and that the error from lossy compression is in the order of these errors. Since no ground truth exists in general, however, this error cannot be quantified but only empirically analyzed. Our experiments have confirmed that lossy compression, as used in our system, does not introduce changes in the overall structure of the computed flow trajectories and that only local changes occur sporadically. On the other hand, our system offers the possibility to use the initial flow fields so that the quality of results can always be verified, yet at the expense of longer computation times.

An additional line of research in turbulence visualization has focused on the extraction and visualization of specific features in such flows. For instance, the dynamics of turbulent near-wall flows have been studied via FTLE-based feature descriptors indicating flow separability [25] to extract so-called splat and anti-splat events in the turbulent boundary layer. Others have shed light on the computation of multiscale turbulence representations and the analysis of the scale-space character of specific features. In turbulence research, this has led to a thorough analysis of the statistical features of velocity increments in the inertial range [26]. To separate scales with features at distinct length scales, filtering techniques have been performed in both the spatial domain via wavelet methods and the frequency domain. In visualization, one line of research is dedicated to the construction of multiscale hierarchies using filtering techniques that are specially adapted to the characteristic length scales of the turbulent features. The optimization-based technique proposed by Lie et al. [27] decomposes the flow field into components at multiple scales related to the captured energy spectrum. Recently, Ngyuen et al. [28] have introduced a modification of the feature level-set method [29] to act as a filter that separates structures at different scales. Feature-based approaches have in common that they require knowing what features to extract and monitor over time. Furthermore, some of these methods are restricted to certain feature types, like critical points in feature flow fields. In this work, we strive for a more general approach that enables interactive analysis of motion fields derived from arbitrary quantities passively or actively advected by the flow.

3 Data Generation and Decomposition

In the following, we describe the dataset we have used in our experiments, and we motivate the particular decomposition of the velocity fields that is used to separate large- and small-scale turbulent superstructures.

3.1 Numerical Simulation

A new direct numerical simulation (DNS) of a turbulent channel flow at a moderately-high friction Reynolds number of $Re_\tau = 1000$ has been performed. $Re_\tau = u_\tau h/\nu$ is based on the friction velocity $u_\tau = \sqrt{\tau_w/\rho}$, the channel semi-height h, and the kinematic viscosity ν, where $\tau_w = \mu \nabla V|_{y=0}$ and ρ are wall shear stress and fluid density, and μ and $V(x,y,z,t) : \mathbb{R}^4 \to \mathbb{R}^3$ are dynamic viscosity and flow velocity, respectively. An open-source, parallel, mixed-discretization solver (publicly accessible via https://github.com/davecats/channel) based on the algorithm described in [30], has been utilized for the simulation. A Fourier–Galerkin discretization is adopted along the wall-parallel directions, while spectral-like fourth-order compact finite differences are utilized along the wallnormal direction (y) [31]. Periodic boundary conditions are utilized along the streamwise (x) and spanwise (z) direction (See Fig. 3a).

The pressure gradient is kept constant during the simulation, allowing for a fluctuating flow rate in the channel. While discretization parameter and numerical methods are standard for such simulations, the peculiarity of the current simulation is the very short streamwise extent of the computational domain(See Fig. 3a). Such small streamwise extent results in the so-called minimal streamwise unit

This article has been accepted for publication in IEEE Transactions on Visualization and Computer Graphics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2022.3232367

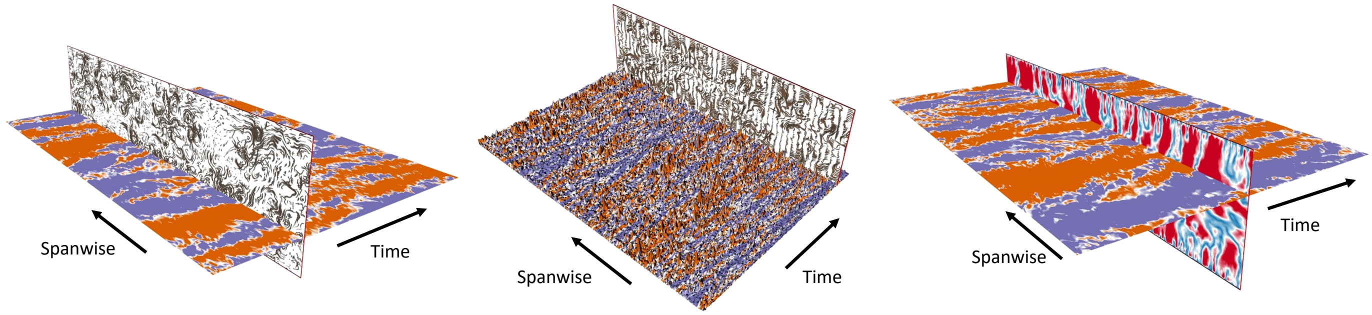

Fig. 2. From left to right: Space-time plot combined with vertical projection plot of 3D pathlines, advanced 2.5D space-time plot and vertical projection plot of streamlines, space-time plot, and visualization of turbulence kinetic energy in the vertical plane.

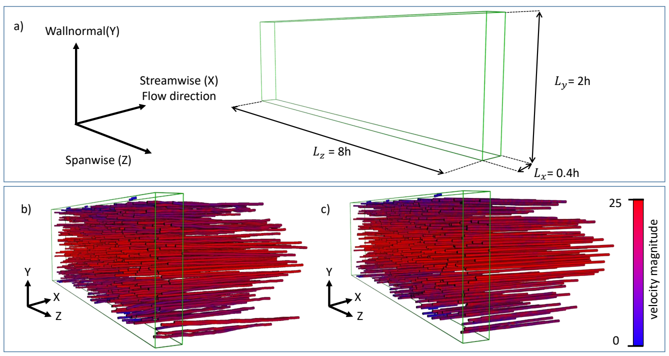

Fig. 3. MSU dataset. (a) The simulation domain. (b) Streamlines in timestep 500, and (c) pathlines from timestep 400 to 1000, of randomly seeded particles. Due to the dominating streamwise velocity component, neither streamlines nor pathlines convey salient features in the flow. Due to periodic boundary conditions, the flow extends outside the 3D channel domain, as indicated by the green wireframe.

(MSU) [32], [33], the computational box with the smallest streamwise extent to still allow the existence of turbulence. The numerical simulation has been carried out over a time span of 0.216 seconds, with time advancing in each timestep of 0.12 ms.

Abe et al. [33] proved that due to the short streamwise dimension of MSU, TSS appear indefinitely long and significantly enhanced in strength by exploiting the domain periodicity. While for a regularly-sized computational domain, a much higher Reynolds number, thus higher computational cost, would be needed to achieve the same strength of large-scale structures. However, not all turbulence statistics in MUS are identical compared to a regular domain as it is discussed in [33]. Therefore, we chose the MSU as a useful model flow to study the interaction between TSS and the near-wall small eddies. Further, The smaller size of the computational box results in a smaller memory required to store a single flow snapshot, enabling us to store 1000 flow fields spaced by $\Delta t = 10\nu/u_\tau^2$ time units, allowing to capture both the slow evolution of TSS and the fast dynamics of near-wall vortices. Considering the additional Fourier modes to prevent aliasing effects, each snapshot is stored on a spatial Cartesian grid with a resolution of $(64 \times 503 \times 2048)$ in streamwise, wallnormal, and spanwise direction, respectively.

In channel flows, the mean streamwise velocity component is relatively large compared to the velocity fluctuations. Fig. 3 shows streamlines in a single timestep of MSU as well as pathlines integrated over 600 timesteps, with the velocity magnitude mapped to color along each line. While higher speed is clearly observed with increasing distance to the channel boundaries, pronounced flow structures can hardly be seen due to the large streamwise velocity component.

3.2 Reynolds Decomposition

It is common in the field of physics of turbulent flows to decompose, according to the so-called Reynolds decomposition, the velocity field V into an average $\overline{V}$ and a turbulent fluctuation u' about this average value, such that $V = \overline{V} + V'$ (see, for instance, [34]). The rationale behind this choice is that the statistics of turbulent flow quantities, such as $\overline{V}$, are deterministic and reproducible, while single realizations of the flow field, i.e., V', are stochastic. Typically, researchers in turbulence are interested in predicting the $\overline{V}$ flow field. Therefore, the effect of velocity fluctuations V' must be physically understood and then modeled. In the present work, we want to assess the interaction between the differently sized flow features in the fluctuation field $\overline{V}$, with particular focus on the large-scale flow structures (see, for instance, [1], [35]). Such large-scale structures are typically defined as sufficiently large regions of V' positive or negative fluctuations. Constrained by this definition, we need to resort to the Reynolds average to identify such large-scale structures.

We have chosen to perform Reynolds averaging only along the streamwise and spanwise directions, yet not in time, so that $\overline{V} = \overline{V}(y,t)$. The reason is that the flow is driven through the channel via a constant pressure gradient so that we can exactly prescribe a priori the intended average value of friction Reynolds number Re_τ. As a result, the flow rate Q will fluctuate in time around its mean value $\overline{Q}$, i.e., there will be instants in time in which the overall flow in the channel is faster and others in which it is slower. By only averaging in the wall-parallel directions, the flow rate fluctuations are included in the average field $\overline{V} = \overline{V}(y,t)$, and thus not considered when the scale properties of V' are studied.

Fig. 4 shows isosurfaces of streamwise velocity component of fluctuation field at a large iso-value (orange) and λ_2-isosurfaces (green). Both surfaces indicate large coherent structures of high velocity close to the channel walls, with some of them reaching into the middle channel part and even connecting themselves.

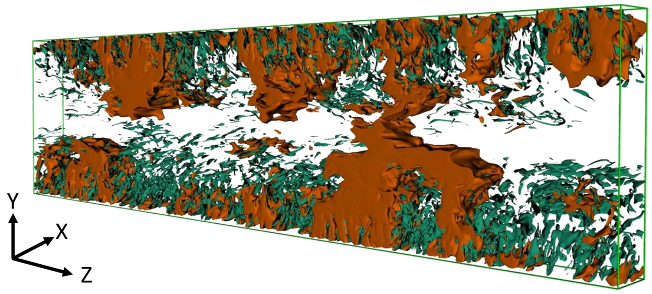

Fig. 4. MSU dataset. Isosurfaces in the streamwise component of the velocity fluctuation field (orange) and in the λ_2 field computed in a single timestep of the velocity fluctuation field (green).

4 Workflow Overview

The proposed visual workflow is designed to enable interactive analysis of the spatial and temporal relationships between multiscale turbulent structures and representative motion structures in 3D unsteady flows. This research objective is addressed through the following analysis tasks:

- Extract multiscale structures and capture the motion of non-material features from the raw flow field. We address this by applying the filtering operator, decomposition of the flow field, and applying the optical flow algorithm to derived scalar feature fields.
- Show 3D spatial structures that correspond to one single row of a space-time plot along with the space-time plot. For that, we introduce the vertical projection plane, which can be moved "in time" over the space-time plot.
- Explore inherent relationships between space-time and particle-based flow features. This is addressed by using classical trajectory-based flow features in various fields.
- Convey spatial proximity to the projection plane, i.e., indicate the spatial location of projected structures with respect to the selected row in the space-time plot. This is accomplished by utilizing color and transparency to fade out structures with increasing.
- Show the temporal dynamics of the spatial structures when moving along the space-time plot in time. Therefore, we provide efficient mechanisms to extract and render the structures in the vertical plane using data compression and GPU computing.

Fig. 5 provides an overview of the proposed workflow. It takes as input a sequence of time-varying 3D vector fields, each given on a Cartesian grid. In the preprocessing phase, sequences of derived fields are computed via Reynolds decomposition. Furthermore, the fluctuation fields are decomposed into high- and low-frequency temporal fluctuations via temporal averaging. By performing image registrations between fields of derived quantities at subsequent timesteps, motion fields indicating the non-apparent advection of these quantities are computed. In the last preprocessing step, the raw and derived fields are compressed using a lossy compression technique with adjustable reconstruction quality.

Next, the user selects a line probe in the spanwise direction at a specific streamwise-wallnormal position, and

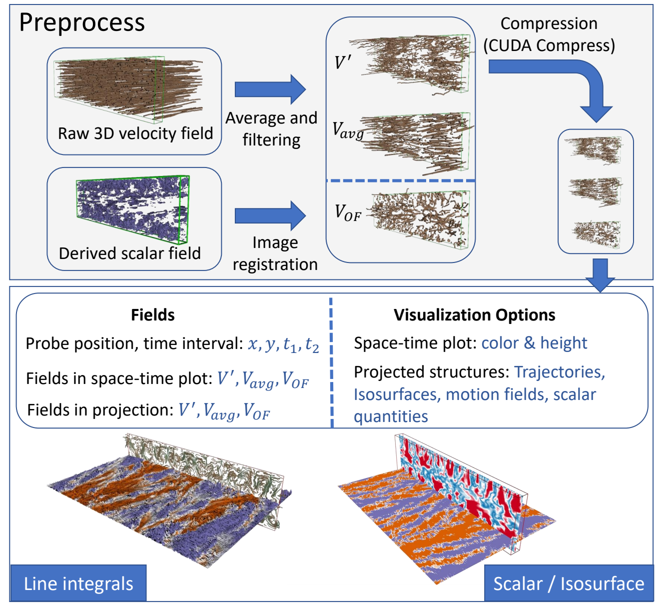

Fig. 5. Workflow overview. In the preprocessing phase, various volumetric fields are computed, compressed, and stored on disk. The lower parts demonstrate some of the interactive options of the visualization scheme, which can be used on the fly to explore the temporal and spatial dynamics of specific flow structures. E.g., x and y, which are the streamwise and wallnormal position of line probe of the space-time.

the field that is considered for the space-time computation. The system automatically computes the space-time fields for all wallnormal positions by sequentially loading all compressed fields to the GPU, decompressing the fields, reading the required values, and storing them into a 3D texture map. Besides the wallnormal position of the line probe, the scale at which the fluctuation field should be displayed can be selected.

Together with the space-time plot, a vertical plane serving as a projection plane for 3D flow structures is indicated by a wireframe. This plane is initially located at the first timestep, and can be moved interactively to an arbitrary time point. Alternatively, it can loop automatically back and forth between a selected start and end time. By default, streamlines in the timestep at which the vertical plane is shown are extracted and projected onto the plane. The user selects the width of the interval along the streamwise direction in which the streamlines are shown. While moving the plane, streamlines are updated automatically. Furthermore, the user can select to show projected pathlines, and streak particles, projected isosurfaces in derived fields, and scalar quantities derived from the current timestep. The extents of the filters that are used to modify the velocities shown in the space-time plot and from which flow features are extracted can be selected to analyze different scales.

4.1 Spatial and Temporal Filtering

Reynolds decomposition is performed by subtracting from the streamwise velocity components at channel height y and time t the plane-wise spatial averages,

$$\overline{u}(y,t) = \frac{\sum_{x,z} u\left(x,y,z,t\right)}{n_x n_z}, \tag{1}$$

This article has been accepted for publication in IEEE Transactions on Visualization and Computer Graphics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2022.3232367

where n_x and n_z are the number of grid points in the streamwise and spanwise direction, respectively, and u is the streamwise velocity component. The resulting field is referred to velocity fluctuation field or V'. Further, as $\overline{u}(y,t)$ changes smoothly between neighboring y positions the derivative of the resulting field is well-defined.

As indicated by the space-time plot (see Fig. 1), the velocity fluctuation field contains both small- and large-scale structures, and these structures behave differently over time. The large-scale structures evolve slowly, both spatially and temporally, unlike the small-scale structures, which vary in comparatively shorter periods and over smaller spatial regions. Using this characteristic space-time behavior, we further segregate the large- and small-scale structures by decomposing the velocity fluctuations into high and low-frequency temporal fluctuations. To accomplish this, temporal filtering is applied to the velocity fluctuation field in addition to spatial filtering. The temporal filtering is performed by applying at every grid point a moving average operator over time:

$$< V(x,y,z,t) >_{w_t} = \frac{1}{w_t}\int_t^{t+w_t} V(x,y,z,\tau)\mathrm{d}\tau \qquad (2)$$

where w_t is the size of the time window, and $V(x,y,z,t)$ is instantaneous velocity vector at position (x,y,z) and time t. The resulting velocity field is referred to as time-averaged velocity fluctuation or V_{avg}. Evidently, large-scale structures are obtained by applying the temporal filter with large w_t. Temporal filtering is, in particular, used to obtain the persistent background features in the velocity fluctuation field.

Fig. 6 shows streamlines in the velocity fluctuation field at timestep 500 in MSU, side by side with the streamlines in the time-averaged fluctuation field where temporal variations are damped out, and only the base structure is kept. Especially in the visualization where streamlines are projected into a plane perpendicular to the wallnormal-spanwise direction, small-scale fluctuations are removed, and the temporally persistent features are emphasized. In all of our experiments, unless otherwise stated, we visualize the velocity fluctuation field in the space-time plot and use the temporally filtered field for extracting additional flow structures via characteristic lines and advection patterns. We used a time window size of $50\Delta t$ to filter out the small-scale feature of the flow without destroying the large-scale structures. The choice of window size is further discussed in Appendix B.

Since the computation of spatial and temporal averages are a one-time relatively high memory and computational consuming step, it is performed in a preprocess before startup time. If the user selects a different setting, e.g., a sliding average using shorter or longer time periods, a waiting time cannot be avoided (see Sec. 4.3).

4.2 Optical Flow

The direct visualization of the turbulent flow, for instance, by means of salient features derived from the velocity gradient tensor or characteristic lines such as streamlines, streaklines, or pathlines, provides a rich characterization of the local quantitative and qualitative behavior of flows and

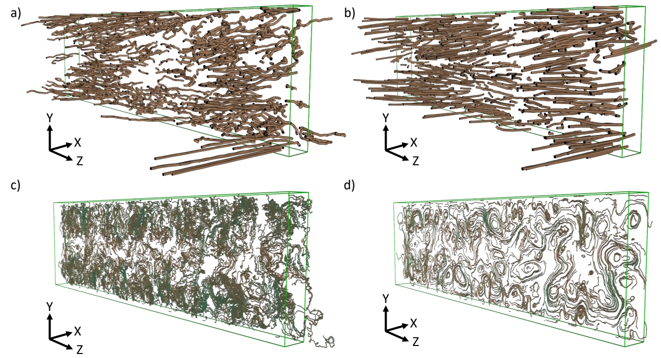

Fig. 6. (a) Streamline visualizations of randomly seeded particles in the velocity fluctuation field, and (b) in the time-averaged velocity fluctuation field of $w_t = 50\Delta t$. (c) and (d) show projections of the streamlines in (a) and (b), respectively, into the wallnormal-spanwise plane.

the existing structures in the flow motion. However, these visualizations do not necessarily present the dynamics of the features of the turbulent flow such as velocity fluctuations at different scales, wall-shear stress, turbulence kinetic energy, or λ_2-isosurfaces, since their advection does not need to be according to the local flow field velocity. In other words, these features are typically not material. While in animations of salient features over time, this motion can be observed, it becomes difficult to determine specific motion patterns as well as their relationship to the movement of TSS over time.

In particular, there are additional advection mechanisms associated to large scale circulation for large-scale turbulent structures which cannot be captured by the local velocity field [32]. Since up to now a correspondence between the temporal evolution of TSS and the motion of structures in the flow field velocity could not be revealed, the conjecture has been raised that these patterns are driven by some of these mechanisms.

For analyzing these relationships, we propose using the so-called optical flow. Given a sequence of 3D time-varying scalar fields, e.g., λ_2 or streamwise component of the velocity fluctuation, the optical flow [36], [37] estimates the apparent motion of the scalar quantities between two successive timesteps. The optical flow computes the displacement field for every spatial location with the associated scalar value to which location this value should be moved such that the field in the first timestep is transformed to the field in the second timestep.

We calculate the optical flow of the derived scalar fields to estimate the motion of the corresponding structures assigned to these fields. For instance, we know that large-scale structures are identified by the magnitude of the streamwise components of the velocity fluctuations. Therefore, the optical flow between two adjacent fields gives us an estimate of the velocities proportional to the motion of the large-scale structures. For optical flow computations we use the toolbox provided by [38], [39] and publicly available at https: bsenneville.free.fr/RealTITracker. A more detailed discussion of the optical flow computation is provided in Appendix A. To experimentally determine the plausibility of the obtained motion fields, we have first pursued a number of experiments using simulated fluid motion fields and scalar quantities which can be considered to some extent,

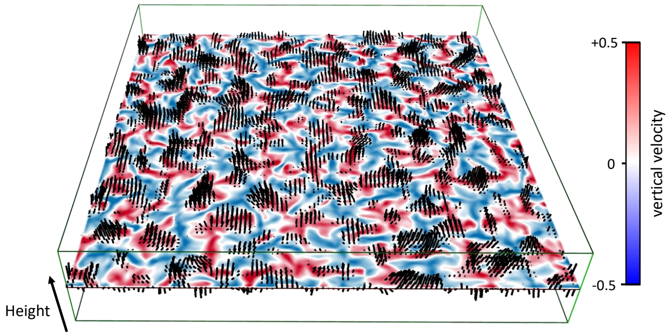

Fig. 7. Visualization of short streamlines regularly seeded in the horizontal middle plane in the optical flow field of two successive snapshots of the time-averaged temperature field in a convective heat flow simulation. In-plane color-coding shows the positive (red) and negative (blue) vertical velocity component of the time-averaged velocity field.

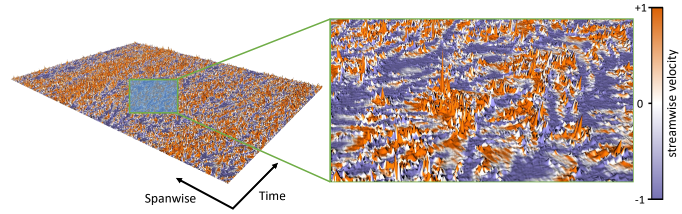

Fig. 8. Space-time plot of streamwise (mapped to color) and wallnormal velocity fluctuation (mapped to height) reveals correlations between the two quantities.

passive tracers in these fields. For instance, we have computed the optical flow between two successive timesteps in the time-averaged temperature field of a Rayleigh-Bénard-type convective heat transport (RBC) simulation to obtain the motion field of the time-averaged temperature field. In this simulation, the air between two parallel planes is heated from the bottom and cooled from the top so that roll-like convection patterns occur due to uprising warmer air and cooled-down sinking air [40]. Fig. 7 shows a visualization of a color-coded plane of vertical velocity in the time-averaged velocity field, combined with short streamlines of particles seeded on a regular grid in the optical flow field of the temperature scalar. One can clearly see the correlation between the two fields, i.e., the motion field of the time-averaged temperature is aligned with the vertical component of the RBC velocity field, which indicates the direction of the convection. Our examples indicate that the optical flow can grasp the overall movement of scalar flow quantities that are derived from the velocity fields.

4.3 Data Management and Compression

In principle, integration-based flow features can be precomputed at a selected seeding density and stored on the GPU. However, even if sufficient GPU memory is available, the user might want to re-seed with a locally increased density, compute the lines in some filtered version of the initial velocity field, or change the time period over which the integration is performed. Under these constraints, alternative mechanisms need to be realized so that the user's flexibility in choosing different settings is not restricted, yet interactivity can still be maintained to a certain extent.

A similar problem arises when computing space-time plots. Pre-computing the space-time plots for each possible streamwise-wallnormal location of the seed line becomes infeasible. On the other hand, domain experts are mainly interested in analyzing the relative changes of the TSS when changing the seed line position along the wallnormal direction. To allow such changes at high performance, we decided to pre-compute the space-time plots for a fixed streamwise position and all grid locations along the wallnormal direction, i.e., for each of these grid locations, a 2D field with the temporal variations of the velocity fluctuations along the corresponding seed line are stored. The resulting 2D space-time fields are stored on the GPU. Once the streamwise location is changed by the user, the corresponding set of 2D fields needs to be re-computed by accessing all timesteps.

To reduce I/O bandwidth limitations when accessing the simulation timesteps, the system is built on top of the publicly available CUDA compression library by Treib et al. [41]. It provides a lossy GPU-based compression scheme using a combination of the discrete wavelet transform, coefficient quantization, and Huffman coding. Decompression is performed on the GPU so that the compressed data can be streamed all the way from the disk to the GPU. We have slightly modified the compression scheme to avoid splitting the data into smaller-sized blocks so that memory address indirections introduced by the use of a block table are avoided. To compress the 3D velocity fluctuation fields from which space-time plots are generated, we use CUDA compress with a fairly aggressive compression ratio of $\simeq 36$. Due to the removal of the mean value, the data range is decreased considerably. Even at this ratio, high reconstruction fidelity is achieved, and the large-scale structures are well maintained (see Fig. 12a). A lower compression ratio of $\simeq 24$ is used to compress the velocity fields in which particle trajectories are computed, i.e., the time-averaged fluctuation fields. Due to the accumulation of reconstruction errors during particle integration, the computed trajectories can be different to those in the initial fields. Treib et al. [42] point out that bounds on the accumulated errors and possible deviations of the computed trajectories cannot be given, yet in an empirical study they demonstrate that the accumulated errors are in the order of the errors that are already introduced by the used integration and interpolation schemes. They further show that the additional errors due to lossy compression do not change the overall structure of the computed trajectories and, thus, do not hinder a phenomenological visual analysis. In our scenario, these effects are even less severe since the characteristic lines are computed in the time-averaged velocity field in which high-frequency variations are damped out to a certain extent. Fig. 12b compares streamlines computed in an initial and compressed fluctuation field and shows that even small-scale geometric details are well preserved. Throughout our experiments, we have never observed that the compression error has yielded any alteration of the large-scale flow structures we aim to analyze in this work. However, since this can not be guaranteed, the system provides the possibility to use the uncompressed flow fields so that the quality of determining results can be verified, yet at the expense of

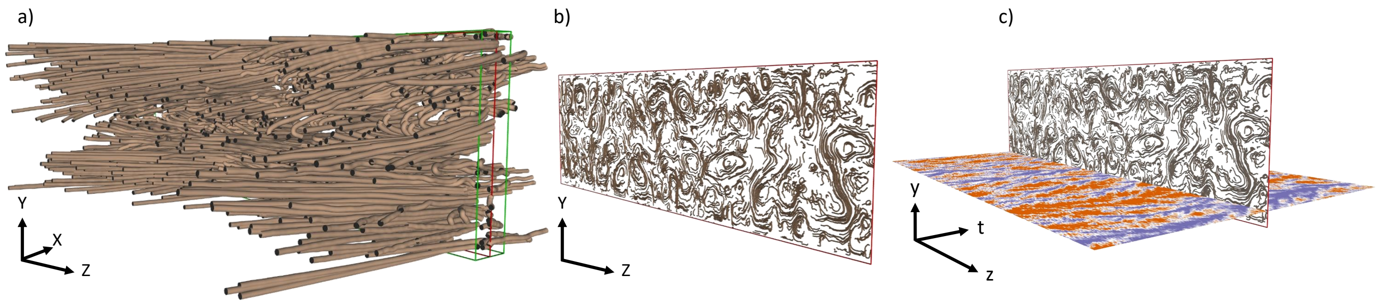

Fig. 9. a) Randomly seeded streamlines in a single timestep of the time-averaged fluctuation field. b) Streamlines are projected onto a selected wallnormal-spanwise plane. c) Combined visualization of space-time plot and vertical projection plot to support visual analysis of the relationships between turbulent structures at multiple scales.

longer computation times.

4.4 Visualization Components

To analyze the spatio-temporal relationships between multiscale streaks and specific patterns in the velocity and motion fields, we combine the Eulerian space-time plot with particle-based flow visualization techniques. An advanced space-time plot enables encoding an additional scalar variable, and by integrating a so-called spatial projection plot, spatial context information regarding the structure of the selected velocity fields is provided.

4.4.1 Advanced Space-Time Plot

For visualizing the 2D space-time plots of velocity fluctuation magnitudes, we use GPU ray-casting on the precomputed 3D texture map. Rays are intersected with the plane in texture space that corresponds to the selected wallnormal position, and the fluctuation values at the ray-plane intersection points are reconstructed using bilinear interpolation. Via a transfer function, the user can select arbitrary mappings of the field values to colors. We decided not to use GPU rasterization to avoid converting the fields into a triangle mesh and leaving the CUDA Compute API for rendering this mesh.

In addition to the standard space-time plot, we introduce an extension that encodes another scalar parameter as height on the space-time plot. The motivation underlying this extension is to investigate the dependencies between the streamwise component of the velocity fluctuations shown by the standard space-time plot and the fluctuations in wallnormal direction. Therefore, the magnitude of the wallnormal component of the velocity fluctuation field is interpreted as height values, and GPU height field ray-casting is performed. Here we employ the method proposed by Dick et al. [43], which projects the rays first into the 2D plane and then performs a discrete differential traversal routine [44] to step from cell to cell in the underlying 2D grid. Within each cell, linear interpolation between the values at the cell entry and exit point is performed to obtain the intersection between a ray and the height field.

We use the advanced space-time plot to further explore the relationships between the streamwise and wallnormal velocity fluctuations (wallnormal component of the velocity fluctuation field) near the wall, i.e., at $0.05h$. Surprisingly, we observe a strong positive correlation between the two velocity components (see Fig. 8), in the form of the absolute magnitude of the wallnormal component and positive streamwise component of the velocity fluctuation, a phenomenon that was so far unknown to the domain experts. Even though no causal dependencies between the up- and downward velocity fluctuations on the one hand and the streamwise component of the velocity fluctuations, on the other hand, can be concluded from this visualization, it suggests investigating further whether specific motion patterns in 3D space and time exist that can explain these mutual relationships in more detail.

4.4.2 Spatial Projection Plot

While the advanced space-time plot shows the full temporal evolution of two quantities along a selected line probe instantaneously, its spatial information is notably limited. In particular, in its current form, the space-time plot does not allow for simultaneous visual analysis of flow quantities in the spatial proximity of the seed line, which is necessary to shed light on the mechanisms driving the evolution of TSS. On the other hand, 3D flow structures around the seed line cannot simply be visualized since the third dimension associated with the streamwise movement is occupied by the time axis of the space-time plot.

We address this limitation by combining the space-time plot with a so-called spatial projection plot. The spatial projection plot shows the projection of 3D flow lines as well as 3D isosurfaces onto the 2D spanwise-wallnormal plane spanned by the selected line probe and the wallnormal direction. Fig. 9 demonstrates the concept underlying this type of visualization, as well as the combination with the space-time plot. The projections into the selected spanwise-wallnormal plane are visualized in a 2D canvas orthogonal to the space-time plot. Since the space-time plot shows the parameter values along the line probe at different times, the spatial projection plot is always anchored at the current timestep, i.e., the canvas moves along the time span. The projected features, as well as their visualization in the canvas, are updated instantly to convey the temporal movement of the plane. Thus, relationships between the features in either view can be revealed by looping back and forth in time (see the accompanying video for an online demonstration). Upon generating the space-time plot, the

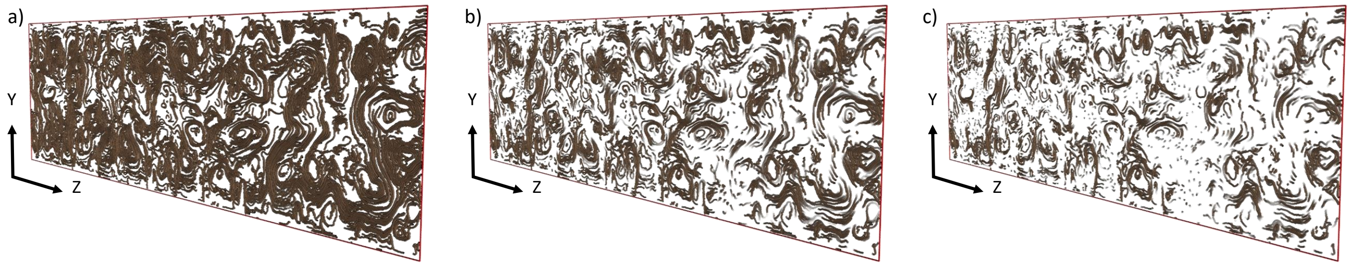

Fig. 10. a) Streamlines are projected into a selected wallnormal-spanwise plane. b) Projected streamlines in (a) are rendered with transparency that depends on their distance to the selected wallnormal-spanwise plane. c) Via a transfer function that controls the assignment of transparency, lines can be faded out more quickly.

user selects one of the following features to be shown in the spatial projection plot:

Streamlines and pathlines: Whenever the user moves the canvas to a new timestep, the flow field at this time is loaded and decompressed. Then, streamlines are integrated forward and backward from random seed positions in the projection plane using the selected number of integration points and stored in 2D buffers on the GPU. Pathlines are computed once by seeding particles in the 3D domain and integrating them through the time-varying flow using the selected time interval. This requires reading and decompressing all velocity fields within this interval. The pathlines are then stored on the GPU in the same way as streamlines.

Visualizing the computed trajectories is performed in the following way: All line primitives are represented by tubes with a user-selected radius. They are rendered into a vertical plane using a parallel projection along the streamwise direction. Correct visibility order is ensured via order-independent transparency (see section 5). Thus, what is seen in the projection plane is a rendering with the camera positioned at infinity and the viewing direction along the streamwise direction. Via a slider in the user interface, the user can adjust the portion of the 3D object space to either side of the projection plane that is considered in the projection plot. All structures outside this portion are clipped during rendering.

The non-injective property of the projection of trajectories leads to a loss of spatial perception. To compensate for this, we propose a visual encoding of the rendered tubes depending on their distance to the projection plane. Note that this plane represents the wallnormal-spanwise plane at the streamwise position of the line probe. Thus, the visual appearance of whatever gets projected into this plane can be modulated depending on the distance from this plane. In particular, the transparency of each fragment, i.e., a surface point when rendering thin tubes, is changed as to rapidly increase with increasing distance to the plane. Fig. 10 shows the effect of using transparency in the spatial projection plot. While the structures in focus, i.e., in or close to the projection plane, can be clearly seen, the surrounding context is only accentuated and, in particular, does not produce severe occlusions of the focus information. When visualizing pathlines in the projection plot, the proposed transparency assignment leads to a smooth transition between the visualized structures when looping through time in an interactive animation. Moving the visualization canvas along the time span can be interpreted as moving a lens along the pathlines, which renders a specific part of the otherwise invisible line visible via increasing transparency.

Scalar quantities: A number of scalar field values like vortex criteria derived from the velocity gradient tensor, e.g., λ_2, wall-shear stress, or velocity magnitude can be selected. Once selected, the velocity field at the current time step is read and decompressed, and the values are resampled onto the current projection plane and displayed. Resampling is performed via ray-casting, i.e., the intersections between the view-rays and the projection plane are computed, and the scalar values are fetched from the loaded timestep. When scalar quantities are computed from a velocity field, this is performed in turn during ray-casting as well. Fig. 13 shows the resulting visualizations for different positions of the vertical projection plane.

Isosurfaces: To indicate the spatial extent of isosurfaces in the before mentioned scalar fields, at each pixel of the projection plane the resampled scalar quantity is compared to a selected isovalue. If the quantity is smaller / larger than the isovalue, a ray is traced forward and backward in streamwise direction (normal to the projection plane) through the 3D scalar field, counting how long the ray stays below (values $<$ isovalue) / above (values $\geq$ isovalue) the isosurface. Fig. 11 demonstrates this for an isosurface in the turbulence kinetic energy field, i.e., $\frac{1}{2}((u')^2 + (v')^2 + (w')^2)$, where u', v', w' are the streamwise, wallnormal, and spanwise components of the velocity fluctuation field, respectively. Both the spatial structure of the surface within the plane and its extent along the spanwise direction are conveyed. This approach, however, does not provide an indication of the topology of the surface along the spanwise direction. For this, a more fine granular characterization of the data values along the rays needs to be performed, along with suitable mappings of these characteristics to color.

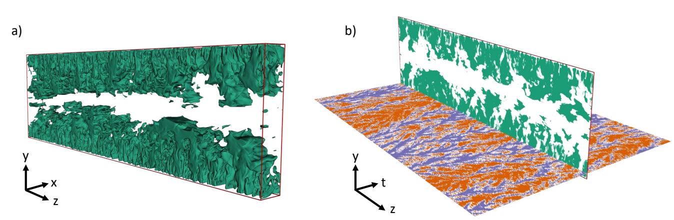

Fig. 11. a) Isosurface in the turbulence kinetic energy of MSU, and b) splace-time plot and projection of the surface onto the projection plane.

While moving the visualization canvas along the time axis sounds intuitive at first, it makes it challenging to capture the changes in the canvas due to its simultaneous movement, i.e., the user needs to adapt to the changes of quantities and position of the canvas at once. To circumvent this drawback, the visualization canvas can be held still, and the space-time plot can be moved "under" it accordingly.

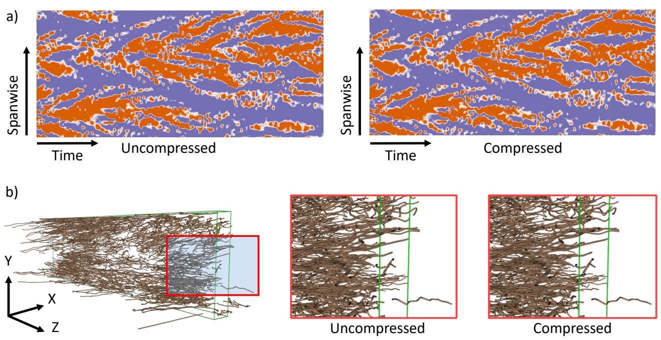

Fig. 12. a) Space-time plots using the uncompressed and compressed streamwise component of velocity fluctuation field. b) Streamlines of randomly seeded particles in the uncompressed and compressed time-averaged velocity fluctuation field. Compression ratios are 32:1 and 22:1, respectively, in (a) and (b).

5 Performance Analysis

All of our experiments are carried out on a system running Windows 10 and CUDA 11.1 with an Intel Core i7-7700K 8x@4.20Ghz CPU, 64GB RAM, and an NVIDIA TITAN RTX. All visualizations are rendered to a 2048x2048 viewport.

For the MSU dataset, consisting of 1000 timesteps of size $64 \times 503 \times 2048$, the initial time to read the uncompressed sequence and to compute the compressed mean-removed velocity fluctuation fields and the time-averaged versions take roughly 10 minutes. This includes storing all compressed data back to disk. The stack of spanwise-time slices, which is required to show the space-time plots for all seed lines at a selected streamwise position and along the wallnormal direction, is stored in a buffer resource on the GPU. Thus, interactive selection of an arbitrary wallnormal position is achieved. When selecting a new streamwise position, the whole flow sequence is streamed and decompressed, and the required data is read to fill the stack. In principle, since only one spanwise-wallnormal slice is read from each timestep, this operation can also be realized on the CPU efficiently by storing a version of the uncompressed sequence where each timestep is laid out spanwise-normal slice by slice. However, we decided not to realize the operation in this way to avoid data duplication.

A single timestep of MSU can be streamed from the CPU (39 milliseconds) and decompressed on the GPU (6 milliseconds) in roughly 45 milliseconds (instead of 740 milliseconds when using the uncompressed data). An even slightly lower time is achieved for the velocity fluctuations due to the higher compression ratio. The computation of 5000 streamlines using a fourth-order Runge–Kutta integration scheme with constant integration stepsize and 500 integration steps requires roughly 9 milliseconds. The computation of 5000 pathlines in MSU over timesteps 500 to 1000 with one integration step per timestep requires roughly 55 seconds, including streaming and decompression of all timesteps. Here we employ the GPU's capability to simultaneously read data from the CPU and work on some other data. Thus, reading/decompression and particle integration can be performed in an intertwined manner, significantly reducing the overall processing time when working on the time-varying data sequence. The integration time interval for pathlines can be selected by the user, for instance, to avoid tracing in early timesteps where turbulence has not yet been fully evolved.

Whenever particle-based line features are computed, per default, the system seeds one primitive every 16th cell in the simulation grid, positioned at the cells' center points. Numerical integration is then performed with the user-selected integration parameters, including the selection of a different seeding strategy. On the GPU, the 3D integration points of streamlines and pathlines are stored in 2D buffers, with one line being stored in every row. Multiple buffers are used if more integration points than entries in one row or more lines than buffer entries in one column are computed. In a typical use case where no more than a few thousand lines are computed and rendered, these buffers are stored uncompressed in GPU memory. If a certain memory budget is exceeded, some buffers are stored in CPU RAM and need to be swapped in for rendering.

Rendering the advanced space-time plot as well as the computed lines requires less than 50 milliseconds in all of our examples. When derived quantities for a single timestep should be rendered, such as λ_2, a scalar feature volume is computed and stored uncompressed on the GPU. In this case, resampling can also be done in less than 10 milliseconds. However, when the user aims to analyze the entire time sequence of a derived feature, the system first loops through the sequence and computes a sequence of derived feature fields. These fields are then compressed using a high compression ratio of 32 and stored on disk. When rendering such a sequence, it takes roughly 50 milliseconds to generate an image of a new timestep.

Line rendering in the spatial projection plot exploits order-independent transparency using per-pixel fragment lists (see [45] for a detailed description of the method). The set of lines is rendered once, with a parallel projection along the streamwise axis, and all generated fragments are stored in a linked list over all pixels. Then, for every pixel, a pixel shader is invoked, which traverses the list and stores all fragments belonging to that pixel in a GPU buffer resource. The fragments are then sorted wrt. their streamwise position. Finally, the fragments determine their transparency depending on the distance to the vertical projection plane and are blended in sorted order. The resulting image is shown in the vertical projection plane. The time required to generate the projection of all lines never exceeded 20 milliseconds in our experiments, where up to 10000 lines were traced.

6 Additional Results

As a case study, we apply the proposed visualization workflow to investigate the relationships between different flow structures at the same and different scales. The user can choose interactively between the multiple pre-computed

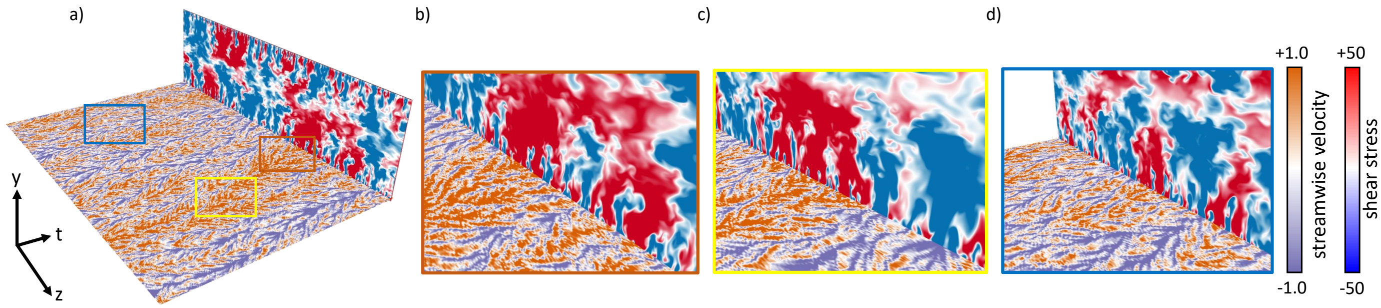

Fig. 13. a) Space-time plot of velocity fluctuations near the boundaries (small-scale turbulent structures) combined with a vertical plane showing the large-scale structures via color-coding of the shear stress on a wallnormal-spanwise plane. In (b) to (d), images of the corresponding cutouts in a) are generated by moving the vertical plane close to regions with strong u'_+ (b) and (c) and weak u'_+ (in d)

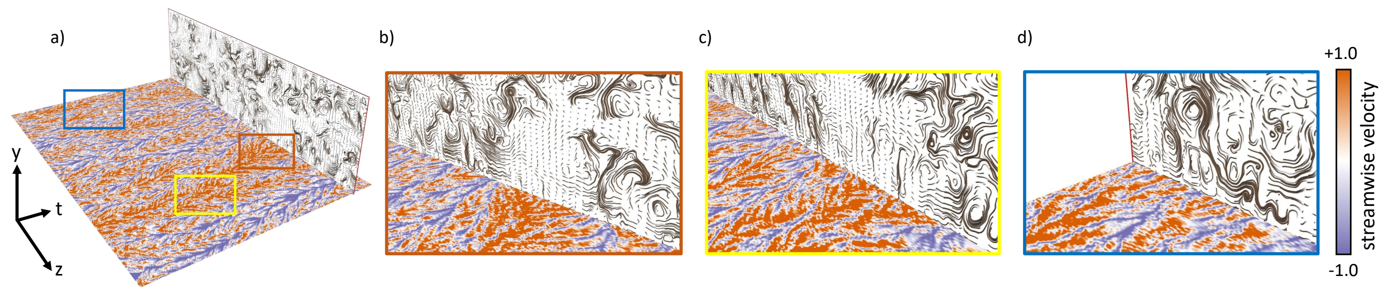

Fig. 14. a) Space-time combined with projection plot, where the space-time depicts the small-scale turbulent structures via showing the streamwise component of the velocity fluctuation near the bottom wall of the channel and the projection plane shows the streamlines of particles seeded on a regular grid in time-averaged velocity fluctuation field with $w_t = 50\Delta t$.The other three figures are generated by moving the projection plot close to regions with strong u'_+ (in b and c) and weak u'_+ (in d)

velocity fields that dominantly contain either large or small-scale structures. For the case of the MSU dataset, the large- and small-scale structures exist dominantly in the time-averaged and velocity fluctuation fields, respectively. In this particular scenario, we use V' in the space-time plot and V_{avg} in the spatial projection plot. The line probe of the space-time plot is positioned close to the wall ($y = 0.05h$) to spot the small-scale structures. Additionally, we eliminate the unwanted low frequencies by first calculating the Gaussian filtered field and subtracting it from the V', which leaves the high frequencies.

Since shear stress, analogous to the streamwise component of the velocity fluctuation, identifies the large-scale structures (see Fig. 13), the TKE is shown on the projection plot to monitor the dynamics of the large-scale structures. The resulting visualization is shown in Fig. 13a. By interactively moving the projection plot to multiple time points, as illustrated in Fig. 13b, c, and d, we see a correlation between the streamwise component of the velocity fluctuation, e.g., blue and orange regions in the space-time plot, and TSS, which can be distinguished as red regions on the projection plane. Moreover, by comparing the regions with stronger and weaker positive streamwise velocity fluctuation (u'_+), we find that the size of the TSS is directly correlated with the magnitude of u'_+. Next, the content of the vertical plane is replaced by a projection plot of the streamlines of regularly seeded particles to investigate the temporal correlation further. We employ transparency to emphasize the streamline segments that reside longer near the line probe. As it is shown in Fig. 14, by moving the line probe in time, one can observe that the regions above strong u'_+ (see Fig. 14b and c) are consistently more sparse compared to regions above weak u'_+. The sparsity can be interpreted as the particles seeded in these regions being moved by relatively laminar flow and transported away from the initial position on the vertical plane at comparably higher speeds. The laminar behavior of the flow in these regions can explain the temporal coherence of large-scale superstructures. On the contrary, the regions above the weak u'_+ are relatively turbulent (see Fig. 14d), which explains the lack of temporal coherence above them.

While the first visualizations reveal the spatial relationships between large- and small-scale turbulent structures, the second set demonstrates their temporal relationships. To further verify our finding, we reproduce the last two experiments for a higher wallnormal position, depicted in Fig. 16. This result confirms the direct spatial correlation between high shear stress and strong u'_+ and the sparsity near the strong u'_+. Additionally, in Fig. 16b we see sparse regions also exists close to the strong u'_-, which confirms further the coherence of large-scale structure, yet with a negative sign. However, further investigations are required to analyze the mechanisms driving TSS and the causal correlations between the multiscale flow structures.

We further shed light on the dynamics of scalar fields such as the streamwise component of the velocity fluctuation, streamwise component of time-averaged fluctuation, turbulence kinetic energy, and λ_2, to examine whether the advection of structures in such fields can explain the causal correlations between the multiscale turbulent structures.

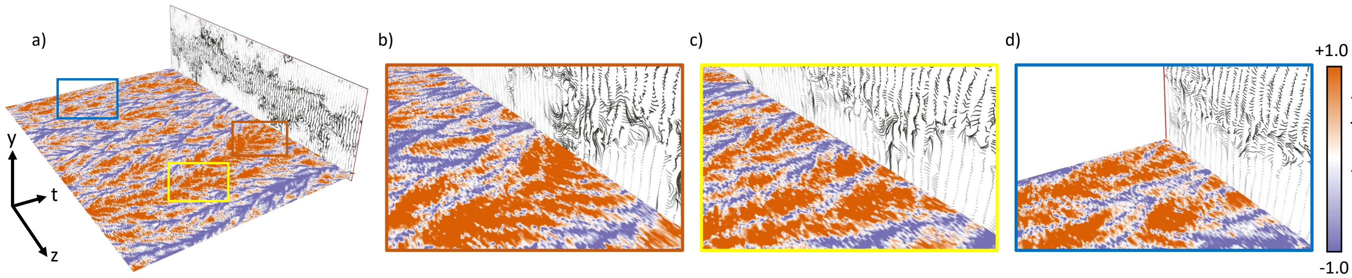

Fig. 15. a) Combined visualization of space-time and projection plot, where the space-time shows the small-scale turbulent structures in velocity fluctuation field and the projection plane depicts the streamlines of particles seeded regularly in the motion field of two consecutive snapshots of streamwise component of the velocity fluctuation field. (b) and (c) depict regions with strong u'_+ and (d) present a region with weak u'_+ and generated by moving (in time) the projection plot to the aforementioned regions.

The temporal changes of the aforementioned scalar fields are captured by the optical flow fields computed between adjacent time steps. In these fields, short streamlines of particles are seeded on a regular 2D grid over the projection plane and combined with a space-time plot of the velocity fluctuation. Fig. 15a depicts an instance of the visualization, where the streamlines are seeded in the motion field of the streamwise component of the velocity fluctuation. Decreasing transparency with increasing distance to the seeding plane is used to enhance structures in spatial proximity.

We further probe various locations with strong and weak u'_+. As it is shown in Fig. 15b, c, the streamlines close to regions with strong u'_+ are relatively dense compared to the regions in the vicinity of relatively small u'_+ (see Fig. 15a).

By using transparency as described, it is shown that the particles seeded in the areas away from strong u'_+ move relatively fast and show more laminar behavior. Fig. 15d shows the motion field near regions with mediocre u'_+, which, in accordance with the former observations, show relatively stronger laminar behavior. These observations suggest that there are advection processes close to the regions with strong u'_+, which may cause the emergence of the TSS.

7 Conclusion and Outlook

We have proposed a novel workflow and visualization techniques for analyzing superstructures at multiple scales in turbulent fluid flows. The workflow has been designed to provide interactive analysis options even for high-resolution unsteady fields by using data compression and GPU-based computation and visualization of streamlines, pathlines, and derived scalar flow quantities. In particular, a novel combination of space-time plots of velocity fluctuation fields with a projection-based visualization of 3D flow structures enables to shed light on the occurrence of turbulent superstructures in their surrounding 3D context. By using the optical flow, also non-advective motion dynamics not aligned with the flow velocity can be revealed.

Our results indicate the potential of the proposed visualization workflow for an interactive visual inspection of the correlations between superstructures at different scales and for analyzing the mechanisms that drive the spatio-temporal formation of these structures. Concerning the latter, we believe that the use of derived motion fields that indicate the spatial and temporal fluctuations of specific flow quantities from an Eulerian point of view can effectively supplement a visual analysis purely based on the instantaneous velocity fields and derived scalar quantities.

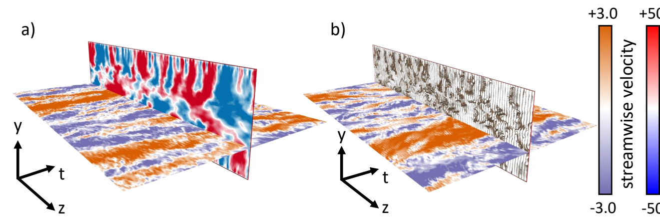

Fig. 16. Space-time plot of streamwise component of velocity fluctuation at $h = 0.7h$ combined with color-coding of shear stress (a) and at $h = 0.8h$ combined with projection plot of the streamlines of particles seeded on a regular grid (b).

The visualization approaches presented in this work have been developed together with collaborators from turbulence research. However, in the current work, we have not yet performed an application study that sheds light on the domain-specific interpretations of the results. We plan to perform such an application study involving different simulation datasets, including superstructures at different lengths and time scales. From a methodological point of view, we will consider augmenting the visualization by linked 3D spatial views, which can offer additional insights into the 3D situation at a selected time point. In particular, we envision using the vertical plane as the view plane for direct volume rendering to simultaneously enhance and suppress, respectively, structures in close proximity and distant to the vertical plane.

Acknowledgments

This work has been done within the Priority Programme Turbulent Superstructures (SPP 1881), funded by the German Research Foundation (DFG).

References

[1] N. Hutchins and I. Marusic, "Evidence of very long meandering features in the logarithmic region of turbulent boundary layers," *J. Fluid Mech.*, vol. 579, pp. 1–28, 2007.
[2] A. J. Smits, B. J. McKeon, and I. Marusic, "High-Reynolds number wall turbulence," *Annu. Rev. Fluid Mech.*, vol. 43, no. 1, pp. 353–375, 2011.
[3] J. Jiménez and A. Pinelli, "The autonomous cycle of near-wall turbulence," *J. Fluid Mech.*, vol. 389, pp. 335–359, 1999.

[4] R. J. Adrian, "Hairpin vortex organization in wall turbulence," *Physics of Fluids*, vol. 19, no. 4, p. 041301, Apr. 2007.

[5] A. Cimarelli, E. De Angelis, J. Jimenez, and C. Casciola, "Cascades and wall-normal fluxes in turbulent channel flows," *J. Fluid Mech.*, vol. 796, pp. 417–436, 2016.

[6] R. Mathis, N. Hutchins, and I. Marusic, "Large-scale amplitude modulation of the small-scale structures in turbulent boundary layers," *J. Fluid Mech.*, vol. 628, pp. 311–337, 2009.

[7] K. M. Talluru, R. Baidya, N. Hutchins, and I. Marusic, "Amplitude modulation of all three velocity components in turbulent boundary layers," *J. Fluid Mech.*, vol. 746, May 2014.

[8] E. Dogan, R. Örlü, D. Gatti, R. Vinuesa, and P. Schlatter, "Quantification of amplitude modulation in wall-bounded turbulence," *Fluid Dyn. Res.*, vol. 51, p. 011408, 2019.

[9] M. Quadrio and P. Luchini, "Integral time-space scales in turbulent wall flows," *Phys. Fluids*, vol. 15, no. 8, pp. 2219–2227, 2003.

[10] J. C. Del Álamo and J. Jiménez, "Estimation of turbulent convection velocities and corrections to taylor's approximation," *Journal of Fluid Mechanics*, vol. 640, pp. 5–26, 2009.

[11] S. Chernyshenko and F. Baig, "The mechanism of streak formation in near-wall turbulence," *Journal of Fluid Mechanics*, vol. 544, pp. 99 – 131, 12 2005.

[12] T. McLoughlin, R. S. Laramee, R. Peikert, F. H. Post, and M. Chen, "Over Two Decades of Integration-Based, Geometric Flow Visualization," in *Eurographics 2009 - State of the Art Reports*, M. Pauly and G. Greiner, Eds. The Eurographics Association, 2009.

[13] R. Bujack and A. Middel, "State of the art in flow visualization in the environmental sciences," *Environmental Earth Sciences*, vol. 79, pp. 1–10, 2020.

[14] M. Schirski, T. Kuhlen, M. Hopp, P. Adomeit, S. Pischinger, and C. Bischof, "Efficient visualization of large amounts of particle trajectories in virtual environments using virtual tubelets," in *Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry*, 2004, pp. 141–147.

[15] K. Bürger, J. Schneider, P. Kondratieva, J. H. Krüger, and R. Westermann, "Interactive Visual Exploration of Unsteady 3D Flows," in *Eurographics/IEEE-VGTC Symposium on Visualization*, K. Museth, T. Moeller, and A. Ynnerman, Eds. The Eurographics Association, 2007.

[16] K. Bürger, R. Fraedrich, D. Merhof, and R. Westermann, "Instant visitation maps for interactive visualization of uncertain particle trajectories," in *Visualization and Data Analysis 2012*, vol. 8294. International Society for Optics and Photonics, 2012, p. 82940P.

[17] A. Frasson, M. Ender, S. Weiss, M. Kanzler, A. Pandrey, J. Schumacher, and R. Westermann, "Visual exploration of circulation rolls in convective heat flows," in *IEEE Pacific Visualization Symposium, PacificVis 2019, Bangkok, Thailand, April 23-26, 2019*, 2019, pp. 202–211.

[18] T. Peterka, R. Ross, B. Nouanesengsy, T.-Y. Lee, H.-W. Shen, W. Kendall, and J. Huang, "A study of parallel particle tracing for steady-state and time-varying flow fields," in *2011 IEEE International Parallel Distributed Processing Symposium*, 2011, pp. 580–591.

[19] H. Childs, E. Brugger, B. Whitlock, J. Meredith, S. Ahern, D. Pugmire, K. Biagas, M. Miller, C. Harrison, G. H. Weber, H. Krishnan, T. Fogal, A. Sanderson, C. Garth, E. W. Bethel, D. Camp, O. Rübel, M. Durant, J. M. Favre, and P. Navrátil, "Visit: An end-user tool for visualizing and analyzing very large data," in *High Performance Visualization–Enabling Extreme-Scale Scientific Insight*, Oct 2012, pp. 357–372.

[20] C. Garth and K. Gaither, "Large-scale integration-based vector field visualization," *Mathematics and Visualization*, vol. 37, pp. 327–338, 09 2014.

[21] J. Zhang and X. Yuan, "A survey of parallel particle tracing algorithms in flow visualization," *Journal of Visualization*, vol. 21, pp. 351–368, 2018.

[22] H. Guo, W. He, S. Seo, H.-W. Shen, E. M. Constantinescu, C. Liu, and T. Peterka, "Extreme-scale stochastic particle tracing for uncertain unsteady flow visualization and analysis," *IEEE Transactions on Visualization and Computer Graphics*, vol. 25, no. 9, pp. 2710–2724, 2019.

[23] M. Treib, K. Bürger, F. Reichl, C. Meneveau, A. Szalay, and R. Westermann, "Turbulence visualization at the terascale on desktop pcs," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 12, pp. 2169–2177, 2012.

[24] T. Günther and H. Theisel, "The state of the art in vortex extraction," *Computer Graphics Forum*, vol. 37, no. 6, pp. 149–173, 2018. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13319

[25] B. Nsonga, M. Niemann, J. Frhlich, J. Staib, S. Gumhold, and G. Scheuermann, "Detection and visualization of splat and antisplat events in turbulent flows," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 11, pp. 3147–3162, 2020.

[26] K. R. Sreenivasan and R. A. Antonia, "The phenomenology of small-scale turbulence," *Annual Review of Fluid Mechanics*, vol. 29, no. 1, pp. 435–472, 1997.

[27] X. Liu, M. Mishra, M. Skote, and C. Fu, "On visualizing continuous turbulence scales," *Computer Graphics Forum*, vol. 38, 2019.

[28] D. B. Nguyen, R. O. Monico, and G. Chen, "A visualization framework for multi-scale coherent structures in taylor-couette turbulence," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 902–912, 2021.

[29] J. Jankowai and I. Hotz, "Feature level-sets: Generalizing isosurfaces to multi-variate data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 2, pp. 1308–1319, 2020.

[30] P. Luchini and M. Quadrio, "A low-cost parallel implementation of direct numerical simulation of wall turbulence," *J Comp Phys*, vol. 211, no. 2, pp. 551–571, 2006.

[31] S. Lele, "Compact Finite Difference Schemes with Spectral-like Resolution," *J Comp Phys*, vol. 103, pp. 16–42, 1992.

[32] S. Toh and T. Itano, "Interaction between a large-scale structure and near-wall structures in channel flow," *J. Fluid Mech.*, vol. 524, pp. 249–262, 2005.

[33] H. Abe, R. A. Antonia, and S. Toh, "Large-scale structures in a turbulent channel flow with a minimal streamwise flow unit," *J. Fluid Mech.*, vol. 850, pp. 733–768, 2018.

[34] S. Pope, *Turbulent Flows*. Cambridge University Press, Cambridge, 2000.

[35] N. Hutchins and I. Marusic, "Large-scale influences in near-wall turbulence," *Phil. Trans. R. Soc. A.*, vol. 365, no. 1852, pp. 647–664, Mar. 2007.

[36] B. K. Horn and B. G. Schunck, "Determining optical flow," *Artificial intelligence*, vol. 17, no. 1-3, pp. 185–203, 1981.

[37] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, ser. IJCAI'81. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1981, p. 674679.

[38] C. Zachiu, B. Denis de Senneville, C. Moonen, and M. Ries, "A framework for the correction of slow physiological drifts during mr-guided hifu therapies: Proof of concept," *Medical physics*, vol. 42, no. 7, pp. 4137–4148, 2015.

[39] C. Zachiu, N. Papadakis, M. Ries, C. Moonen, and B. D. de Senneville, "An improved optical flow tracking technique for real-time MR-guided beam therapies in moving organs," *Physics in Medicine and Biology*, vol. 60, no. 23, pp. 9003–9029, nov 2015. [Online]. Available: https://doi.org/10.1088/0031-9155/60/23/9003

[40] A. Frasson, M. Ender, S. Weiss, M. Kanzler, A. Pandey, J. Schumacher, and R. Westermann, "Visual exploration of circulation rolls in convective heat flows," in *2019 IEEE Pacific Visualization Symposium (PacificVis)*, 2019, pp. 202–211.

[41] M. Treib, "Gpu-based compression for large-scale visualization," Ph.D. dissertation, Technische Universität München, 2014.

[42] M. Treib., K. Brger., J. Wu., and R. Westermann., "Analyzing the effect of lossy compression on particle traces in turbulent vector fields," in *Proceedings of the 6th International Conference on Information Visualization Theory and Applications - IVAPP, (VISIGRAPP 2015)*, INSTICC. SciTePress, 2015, pp. 279–288.

[43] C. Dick, J. Krger, and R. Westermann, "GPU Ray-Casting for Scalable Terrain Rendering," in *Eurographics 2009 - Areas Papers*, D. Ebert and J. Krger, Eds. The Eurographics Association, 2009.

[44] J. Amanatides and A. Woo, "A Fast Voxel Traversal Algorithm for Ray Tracing," in *EG 1987-Technical Papers*. Eurographics Association, 1987.

[45] M. Kern, C. Neuhauser, T. Maack, M. Han, W. Usher, and R. Westermann, "A comparison of rendering techniques for 3d line sets with transparency," *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2020.

Behdad Ghaffari received a M. Sc. in applied and engineering Physics from the Technical University of Munich and currently works as a PhD student in the chair of Computer Graphics and Visualization at TUM. His research interests include time-dependent data visualization, real-time rendering of large data and flow visualization.

Davide Gatti Davide Gatti studied aeronautical engineering at Politecnico di Milano (Italy) and received his Ph.D. in mechanical engineering from the Technical University of Darmstadt (Germany). Since 2017 he is a permanent researcher and lecturer at the Karlsruhe Institute of Technology. His research interests include the physics and control of turbulent flows, turbulence modelling, high-performance computing and simulation of turbulent flows.

Rüdiger Westermann studied computer science at the Technical University Darmstadt and received his Ph.D. in computer science from the University of Dortmund, both in Germany. In 2002, he was appointed the chair of Computer Graphics and Visualization at TUM. His research interests include scalable data visualization and simulation algorithms, GPU computing, real-time rendering of large data, and uncertainty visualization.