Computer Graphics Forum, Volume 31, Issue 1, 2019, pp. 300-315
DOI:10.1111/cgf.13532

Volume 0 (1981), Number 0 pp. 1–15 COMPUTER GRAPHICS *forum*

On Visualizing Continuous Turbulence Scales

Xiaopei Liu[1], Maneesh Mishra[2], Martin Skote[3] and Chi-Wing Fu[4]

[1]ShanghaiTech University
[2]Nanyang Technological University
[3]Cranfield University
[4]The Chinese University of Hong Kong

Abstract

Turbulent flows are multi-scale with vortices spanning a wide range of scales continuously. Due to such complexities, turbulence scales are particularly difficult to analyze and visualize. In this work, we present a novel and efficient optimization-based method for continuous-scale turbulence structure visualization with scale decomposition directly in the Kolmogorov energy spectrum. To achieve this, we first derive a new analytical objective function based on integration approximation. Using this new formulation, we can significantly improve the efficiency of the underlying optimization process and obtain the desired filter in the Kolmogorov energy spectrum for scale decomposition. More importantly, such a decomposition allows a "continuous-scale visualization" that enables us to efficiently explore the decomposed turbulence scales and further analyze the turbulence structures in a continuous manner. With our approach, we can present scale visualizations of direct numerical simulation data sets continuously over the scale domain for both isotropic and boundary layer turbulent flows. Compared with previous works on multi-scale turbulence analysis and visualization, our method is highly flexible and efficient in generating scale decomposition and visualization results. The application of the proposed technique to both isotropic and boundary layer turbulence data sets verifies the capability of our technique to produce desirable scale visualization results.

1. Introduction

Turbulent flows are multi-scale in nature. As revealed by both experimental observations and numerical simulations, the vortical structures in turbulent flows can span a wide and continuous range of spatial scales [Pop01]. Since vortical structures of many different scales could co-exist collectively in one turbulent flow, it is thus complicated to analyze and visualize turbulence dynamics directly from the turbulence data sets.

This motivates the development of multi-scale methods [Far92, BMP08] to analyze turbulent flows by scale decomposition. For example, finer-scale flow structures tend to be more stretched compared to the coarser-scale structures [BMP08], but such a phenomenon is not obvious when directly examining the original flow, which aggregates all different scales. Moreover, analyzing the inter-scale transition and interaction of flow structures by multi-scale decomposition can enrich our understanding of turbulence dynamics, especially for the transition from laminar to turbulent flows in boundary layers [YP11]. However, current methods that rely on wavelets [Far92, vdB04] and curvelets [BMP08, MHVD09] for multi-scale turbulent flow decomposition only produce discrete scales that are limited by the data resolution, preventing useful analysis on transitional flows. For such purpose, continuous-scale decomposition, where scales continuously transit over the spectrum, is more favorable [MLSF14], and the visualization of continuously varying spatial scales is highly desirable for the scientific study of turbulent flows, from both the fundamental perspective [Jim13] and the modeling purposes [Spa15].

Scale decomposition for turbulent flows is different from the conventional vector field decomposition in that it often employs the Kolmogorov energy spectrum [MHVD09] [YPI10] to characterize the decomposed scale [SDT06]. It was discovered earlier by Kolmogorov that the energy of turbulent flows follows a similar spectrum, especially for the inertial range, where the famous $5/3$-law was derived and verified [Kol41, Kol62]. Such a spectrum indicates the variation of turbulent flow energy in different length scales. A *scale* in the decomposition is usually referred to as a band of wavenumbers in the Kolmogorov energy spectrum. Ideally, this should be determined by employing a perfect band-pass filter (BPF) in the spectrum. However, such a straightforward decomposition usually leads to strong ringing artifacts [Gib99], which contaminate the decomposition and disturb the real turbulence scale structures in the decomposition results. This is why wavelets- and curvelets-based methods are often employed for the decomposition task.

While wavelets and curvelets can better represent the data with reduced ringing artifacts due to their multi-scale basis functions, they have various drawbacks that make them inappropriate for continuously decomposing high-resolution turbulent flows: finite number of scales that are restricted by the data dimension, relative-

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd. Published by John Wiley & Sons Ltd.

Published by Wiley. This is the Author Accepted Manuscript issued with: Creative Commons Attribution Non-Commercial License (CC:BY:NC 4.0).
The final published version (version of record) is available online at DOI:10.1111/cgf.13532. Please refer to any applicable publisher terms of use.

ly large memory footprint, and high computational cost. Recently, Mishra et al. [MLSF14] proposed a new method called KoSCO based on filter optimization to operate directly in the domain of the Kolmogorov energy spectrum. It was the first work to overcome the drawbacks of wavelets- and curvelets-based methods with comparable spectrum distribution for continuous turbulence scale decomposition. However, its computational cost is still very high due to the discrete numerical integral of the objective function, making it inefficient for supporting analysis and visualization of large-scale time-varying turbulent flows. In addition, this work explored only isotropic turbulent flows, but not other situations such as boundary layer turbulent flows, which behave quite differently.

Our approach. To address the drawbacks of existing methods, we present in this paper a novel method, aiming at improving the computational efficiency of continuous-scale turbulent flow decomposition for flow visualization in large-scale turbulence data sets. By adopting the framework of [MLSF14], our contribution lies in a *new mathematical formulation* that delivers a very efficient continuous-scale turbulence visualization. This includes a reformulation of the objective function with integral approximations to transform the original discrete numerical integral into a purely analytical one, as well as the pre-computation of the optimization for the analytical objective function, allowing us to significantly reduce the heavy computational cost. More importantly, the decomposition results in a new style of visualization, which combines isosurface technique for the spatial structure of the decomposed scale and the surface texturing technique by referring to the color-mapping of the corresponding velocity field. Hence, we can more effectively and intuitively visualize and analyze the high-resolution continuous-scale decomposition data sets for turbulent flows.

Based on our scale decomposition method with an analytical objective function, we can efficiently generate continuous-scale visualization results. Instead of focusing on the visual analysis of single-scale local/global features as in existing visualization works such as [SUT05, SBV*11], we can present visualization results across a continuous range of scales for both isotropic and boundary layer turbulent flows. This is highly beneficial for the scientific study of turbulence structures, and has not yet been explored and presented in any previous flow visualization works. In addition, experts in turbulence research are also involved as collaborators in this work to support with domain knowledge, and to help evaluate and enhance the visualization results.

2. Related Work

Turbulent flow analysis and visualizations are usually based on direct numerical simulation (DNS) data sets, which are useful, e.g., in studying near-wall flow structures [WM09] and developing new turbulence models [RPP11], where accurate experimental measurements are still difficult to obtain. Moreover, the control of turbulence with methods yet to be realized in a laboratory can also be investigated by using DNS [Sko13, Sko14]. This section reviews and discusses the areas of research related to turbulent flow analysis and visualization based on DNS data sets. Note that due to space limits, we do not include research work in general flow simulation and visualization.

2.1. Multi-scale Analysis of Turbulence

A key approach to turbulence analysis is multi-scale flow decomposition. Early methods [Far92, CD99] are mainly wavelet-based since wavelets are multi-scale in nature and can help reduce the ringing artifacts. However, wavelets have a number of shortcomings; e.g., Candes and Donoho [CD99] showed that due to the isotropic basis, wavelets are not effective in representing stretched structures in a turbulent flow. Hence, curvelets, which are extensions of wavelets with elongated basis, were proposed [CDDY06] for scale decomposition with highly-stretched structures. Bermejo-Moreno and Pullin [BMP08] presented a multi-scale geometrical decomposition method based on curvelets for isotropic flows while Yang et al. [YPI10] showed the evolutionary geometry of the Lagrangian scalar field for stationary isotropic turbulence. Other than wavelets and curvelets, Leung et al. [LSD12] used a spatial filter for flow decomposition, but since the filter is too wide, the decomposed flow may include excessively many nearby scales. Recently, Mishra et al. [MLSF14] developed an optimization method to construct Fourier-space filters, but their method is inefficient and computationally less reliable for continuous-scale flow decomposition.

This paper presents a new continuous-scale decomposition method based on optimization. Compared to previous works, we derive a novel analytical model for the objective function, as well as various techniques for supporting efficient and stable decomposition of turbulent flows in continuous scales. Thus, we can deliver turbulence scale structure visualizations and support visual analysis for high-resolution flow data sets in various situations.

2.2. Turbulence Visualization

There are three major research directions in turbulence visualization: i) improving the data processing efficiency, particularly for interactive visualization; ii) developing visualizations targeted for specific applications; and iii) enhancing the visualization by identifying structures/features in the flow data sets.

Among various works in the first research direction, Johnson et al. [JCG08] developed a system that optimizes the data management and caches the computations, thereby enabling interactive visualization of terabyte-sized flow data sets. Treib et al. [TBR*12] presented a GPU-based system design for feature-based turbulence visualization; their method works on a flow field with a compressed representation, and can efficiently deliver high-resolution visualization on a desktop computer.

The second research direction is application-oriented. Wiebel et al. [WTS*07] employed the footprint of vortices induced from boundary walls to form a new type of streak line visualization. Williams et al. [WPB*11] used a reference model of an ideal vortex to model and identify real vortex cores for geophysics. Wei et al. [WYG*11] introduced a dual-space method to analyze particle data from turbulent combustion simulation using model-based clustering. Koehler et al. [KWDG11] developed visual analysis of vortices produced from the deformable flapping wings of a dragonfly. More recently, Shafii et al. [SOL*13] extracted vortices in wind farms, and visualized and analyzed the interplay between these vortices and the forces on the wind turbine blades.

The third direction focuses on extracting turbulent features for

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

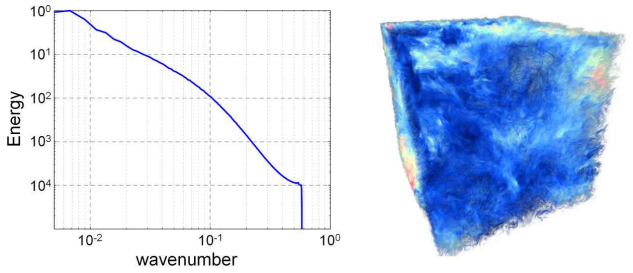

Figure 1: *The Kolmogorov spectrum (left) of a* 512^3 *DNS data set (right) of isotropic turbulence. The color in the direct volume rendering of the isotropic turbulent DNS data set indicates the magnitude of the velocity field with red colors indicating high velocities and blue colors representing low velocities.*

turbulence visualization, where many criteria have been explored, e.g., the Q and λ_2 criteria [JH95]. Silver and Wang [DX97] isolated and tracked the local volume-based features in the form of regions-of-interest in time-varying 3D fluid data sets. Stegmaier et al. [SUT05] combined vortex core line detection and the λ_2 method to visualize and analyze turbulent flows. Helgeland et al. [AØA*04] visualized the energetic structures in a turbulent flow field by using structure-based tensors. Later, Helgeland et al. [ABAPØCE07] developed a vorticity field line approach with specialized particle advection and a seeding strategy. Schafhitzel et al. [SVG*08] improved the vortex core line detection by constructing curves that connect λ_2 minima. Pobitzer et al. [PTA*11] employed proper orthogonal decomposition (POD) to separate the energy scales in a turbulent flow.

Coherent structures are another useful features for enhancing turbulence visualization. Garth et al. [GGTH07] characterized and visualized coherent Lagrangian structures by adaptive computation of finite-time Lyapunov exponent fields. Schafhitzel et al. [SBV*11] visualized and tracked coherent structures based on shear stress, so that one can identify and track both vortices and shear layers, while Gaither et al. [GCS*12] detected and visualized critical structures in a massive turbulent-flow simulation.

This work presents a continuous and highly efficient scale decomposition method to aid turbulent flow visualizations. Compared to previous works in turbulent flow visualization and analysis, we devise a novel optimization-based technique in the Kolmogorov energy spectrum. Due to our analytical formulation and various related techniques, we can efficiently obtain the turbulence scales with high performance. Through this, we can efficiently optimize the separation of turbulence scale structures, and then extract and visualize useful features relevant to the Kolmogorov energy spectrum. This is a new form of decomposition-based visualizations, where we developed the research work with domain experts in turbulence research, and explored several data sets, including high-resolution isotropic and boundary layer turbulent flows.

3. Our Approach

For visualizing continuous turbulence scale structures, we focus our method on an efficient continuous-scale decomposition, followed by an isosurface technique with surface texturing to help visualize and analyze various scales in the turbulence data sets in a continuous manner.

3.1. Efficient Continuous Scale Decomposition

This section presents our novel optimization formulation to efficiently decompose a turbulent flow field based on the Kolmogorov energy spectrum. Such a spectrum is computed based on the Fourier transform of a turbulent flow field, and is one-dimensional by nature [Pop01], see Figure 1 for an example.

To achieve our goal, we propose the followings: first, we describe how to compute the Kolmogorov energy spectrum (subsection 3.1.1) and employ a parameterized filter shape with fall-off regions in the spectrum domain similar to the filter in [MLSF14] (subsection 3.1.2). Then, we formulate an objective function to measure the filter sharpness and the amount of ringing (subsection 3.1.3), with a novel derivation to give an efficient analytical model of the objective function with simplification and approximation (subsection 3.1.4). Finally, we solve the optimization to obtain appropriate filters for the decomposition (subsection 3.1.5).

3.1.1. Kolmogorov Energy Spectrum

The Kolmogorov energy spectrum is a one-dimensional function that specifies the energy distribution over scales in a turbulent flow field. To obtain such a spectrum, one usually first calculates the Fourier transform of the velocity magnitude, and then integrates all the Fourier coefficients whose norms are identical. This projects the multi-dimensional spectrum in Fourier space into a one-dimensional spectrum. To have a better observation, the log-log plot of the spectrum is usually preferred, and the famous Kolmogorov 5/3-law was also discovered in such a log-log plot of the spectrum [Pop01].

With the Kolmogorov energy spectrum, the turbulence scales are usually defined with a filter in the log-log plotted space. To define such a filter, we work in the Kolmogorov energy spectrum with a logarithmic mapping, which we call the *log-mapped Kolmogorov energy spectrum*. Note that such a process can be reversible. Once we have a filter in the log-mapped Kolmogorov energy spectrum, we can obtain the filter in Fourier space by first performing an inverse logarithmic mapping of the filter and then symmetrically extending the re-mapped 1D filter to form a multi-dimensional filter, assuming that the wave vectors are equally distributed along the iso-sphere. Finally, the original flow field is multiplied by the filter in Fourier space, followed by an inverse Fourier transform to obtain the decomposed scale. Note that to obtain the desired decomposed scale, the distribution of the filter in the log-mapped Kolmogorov energy spectrum is important.

3.1.2. Parameterized Band-Pass Filter

Our filter is specified in the log-mapped Kolmogorov energy spectrum domain with a variable k in that range. Note that when k varies, it corresponds to a non-linear variation with respect to an inverse logarithmic mapping in the Kolmogorov energy spectrum. Similar to [MLSF14], we can build our one-dimensional parameterized band-pass filter (BPF) by attaching two extended fall-off

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

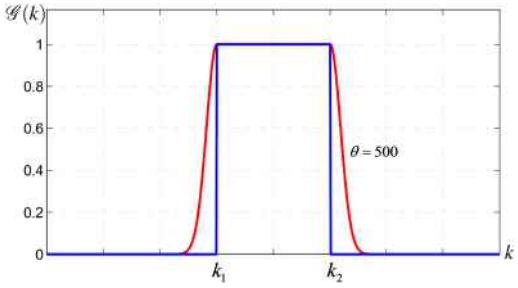

Figure 2: *Blue line: perfect band-pass filter (BPF). Red lines: our parameterized filter with two extended fall-off regions.*

regions to a perfect BPF, see Figure 2. Note that to avoid bias, the two fall-off regions should be symmetric. Moreover, they should be monotonically attenuating in both spatial and wavenumber domains, i.e., the filter value should gradually drop to zero, e.g., see the red curves in Figure 2, to avoid unwanted intermittency effects.

Here we use the Gaussian function with a shape parameter θ to model the fall-off regions in our parameterized filter, and the whole filter is defined in the log-mapped Kolmogorov energy spectrum domain as:

$$\mathcal{G}(k;\theta) = \begin{cases} 1 & \text{if } k \in [k_1, k_2] \\ e^{-\theta(k-k_1)^2} & \text{if } k < k_1 \\ e^{-\theta(k-k_2)^2} & \text{if } k > k_2 \,, \end{cases} \tag{1}$$

where $\mathcal{G}(k)$ is defined in the log-mapped Kolmogorov energy spectrum domain; k_1 and k_2 are lower and upper bounds of the filter band, respectively; and θ is a parameter to control the extent of the fall-off. By varying θ, we can introduce different amount of nearby scales into the decomposition to suppress ringing. However, we need to keep θ small to maintain the band-pass property for the decomposition. Note that when we do the actual filtering, we should transform the filter back to the Fourier domain with wave vector $\mathbf{k}$, where $k = logmap(|\mathbf{k}|)$ with $logmap(\cdot)$ the operator mapping from linear to logarithmic space, and we further denote the filter in Fourier space as $G(\mathbf{k})$. Note that k is normalized in the range $[0,1]$, while $\mathbf{k}$ is kept with its original range in Fourier space.

The reason why we choose the Gaussian function is due to the minimization property of the uncertainty principle [Str08]: when the support range of a Gaussian in the spatial domain is squeezed for suppressing the ringing, its spreading in the wavenumber domain is simultaneously minimized. Thus, we can employ a Gaussian to look for a balance between the support in the spatial domain and the filter sharpness in the wavenumber domain.

3.1.3. The Objective Function

After defining the filters in the log-mapped Kolmogorov spectrum domain, we next reformulate the objective function as in [MLSF14] for finding suitable filters to decompose the turbulence scales by optimization. To facilitate our discussion and subsequence derivations, we first define the following quantities:

- $\mathbf{u}$ and u: the flow velocity field and its magnitude in spatial domain;
- $\mathbf{k}$ and k: the wave vector and its log-mapped magnitude ($k \in [0,1]$);
- $\mathcal{G}$: the desired filter in the log-mapped Kolmogorov energy spectrum domain;
- $\mathcal{G}_0$: the perfect BPF ($\theta \to \infty$) in the log-mapped Kolmogorov energy spectrum domain;
- G: the desired filter in Fourier space, which is a function of the wave vector $\mathbf{k}$;
- hat ($\hat{}$): the operator to transform from spatial domain to the log-mapped Kolmogorov energy spectrum domain;
- hat ($\check{}$): the inverse operator of ($\hat{}$) to transform from the log-mapped Kolmogorov energy spectrum domain to spatial domain;
- θ_l and θ_u: the lower and upper bounds of θ, respectively, during the optimization process; and
- θ_m: the optimal θ value after optimization.

The formulation of the objective function in this paper follows the framework in [MLSF14]. Hence, we first review the objective function in [MLSF14] before deriving our analytical form of the objective function.

Filter sharpness. The first term E_d of the objective function evaluates the filter sharpness by measuring how close the decomposition results by $\mathcal{G}$ and $\mathcal{G}_0$ are, which is defined in the log-mapped Kolmogorov energy spectrum domain as:

$$E_d = \psi_d \int_0^1 (\mathcal{G}(\theta) - \mathcal{G}_0)\, \mathrm{d}k \,, \tag{2}$$

where ψ_d is formulated as $\psi_d = \left(\int_0^1 (\mathcal{G}(\theta_l) - \mathcal{G}_0)\, \mathrm{d}k\right)^{-1}$, which normalizes E_d to $[0,1]$. Note that the support range of $\mathcal{G}(\theta)$ is always larger or equal to $\mathcal{G}_0$. Hence, there is no need to square the integrand in Eq. 2.

Amount of ringing. Recall that the Fourier basis has global support, and it propagates wave-like rings from high-frequency fluctuations, thus contaminating surrounding smooth regions. Hence, we estimate the amount of ringing to constrain the flow decomposition by examining the difference between the given and the decomposed flow fields in the originally-smooth regions in spatial domain Ω:

$$E_r = \psi_r \int_\Omega \left| w \nabla \left(u - \check{\mathcal{G}}(\theta) * u \right) \right|^2 \mathrm{d}\Omega, \tag{3}$$

where $\psi_r = \left(\int_\Omega \left| w \nabla \left(u - \check{\mathcal{G}}(\theta_u) * u \right) \right|^2 \mathrm{d}\Omega \right)^{-1}$ is a normalization factor similar to ψ_d; $w = e^{-|\nabla u|^2 / 2\sigma^2}$ is a Gaussian weight with respect to the gradients of the original field, helping to enforce a strong constraint in the originally smooth regions; and σ is the standard deviation of the gradients in the whole original flow field. Here, by saying "originally smooth", we mean the smooth regions (with small gradients) in spatial domain from the original (given) flow field without any operation. Note that while E_d is defined in the log-mapped Kolmogorov energy spectrum domain, E_r is, however, defined in the spatial domain.

Eqs. 2 and 3 can be used to construct an objective function, but they are not efficient for computation since they are defined in two different domains which are connected by Fourier transform. Moreover, their current formulations cannot lead to effective parameter

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

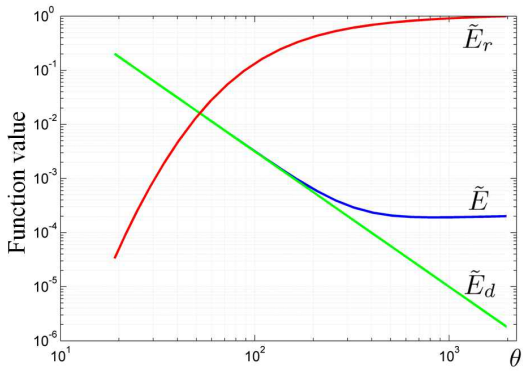

Figure 3: *Objective function $\tilde{E}$ and its constituting components $\tilde{E}_d$ and $\tilde{E}_r$. Note that both axes are in log-scale.*

fine-tuning for creating a filter with a sharp fall-off. Hence, we enhance the order of magnitude of their measurements, and raise both E_d and E_r by an exponent m, which is set to be 5 in our implementation, in order to match the decomposition result by curvelets as presented in [BMP08].

Overall objective function. Based on the above formulations, we define the overall objective function E as a linear combination of E_d and E_r with parameter $\lambda > 0$ to control the influence of E_r to E:

$$E = {E_d}^m + \lambda {E_r}^m, \tag{4}$$

Note that a larger λ introduces more nearby scales to the decomposition result, thus better suppressing the ringing artifacts; however, the filter shape ($\mathcal{G}$) in return would deviate more from the perfect band-pass filter ($\mathcal{G}_0$).

3.1.4. Deriving an Effcient Analytical Form

Eq. 4 is the objective function in [MLSF14]. Since it requires a numerical integration over the entire discrete data space, it is computationally very expensive to evaluate. Especially, it would require a large number of iterations to complete the optimization when processing high-resolution 3D DNS data sets. Having noted that the model can be reformulated by considering domain transformation and integral approximation, which is our core technical motivation, we derive and reformulate Eq. 4 to form an analytical objective function, which can be evaluated with exceedingly high performance. Note that such a mathematical derivation and reformulation has never been done in any existing work.

Domain consistency. Since E_d and E_r are defined in two different domains (E_d in wavenumber domain and E_r in spatial domain), their optimization becomes inefficient due to the time-consuming integral transforms. Hence, like in [MLSF14], Plancherel's theorem [Pin02] is employed to unify E_d and E_r by transforming E_r from the spatial domain into the Fourier domain:

$$E_r = \psi_r \int_W |\hat{w} * i\mathbf{k}\hat{u}(1 - G(\mathbf{k};\theta))|^2 \, \mathrm{d}\mathbf{k}, \tag{5}$$

where $\psi_r = \left(\int_W |\hat{w} * i\mathbf{k}\hat{u}(1 - G(\mathbf{k};\theta_u))|^2 \, \mathrm{d}\mathbf{k} \right)^{-1}$. Note that to compute the integral, we need to transform from the log-mapped Kolmogorov energy spectrum domain back to the Fourier domain with symmetric extension of k to form wave vector $\mathbf{k}$ as explained in Section 3.1.1.

Reformulating E_d. The reformulation of E_d term is relatively easy. Since we normalize k in the range $[0,1]$ and the filter is defined in this range, in order to obtain an analytical form, we can obtain an integral approximation of Eq. 2:

$$E_d = \psi_d \int_0^1 (\mathcal{G}(\theta) - \mathcal{G}_0) \, \mathrm{d}k \simeq \psi_d \int_{-\infty}^{+\infty} (\mathcal{G}(\theta) - \mathcal{G}_0) \, \mathrm{d}k. \tag{6}$$

The approximation is still accurate, since the function values outside $[0,1]$ is relatively very small compared to function values in $[0,1]$. Inserting the specific form of $\mathcal{G}$ (Eq. 1) into Eq. 6, we can reformulate E_d as

$$\tilde{E}_d = \sqrt{\frac{\theta_l}{\theta}}, \quad \text{where } \theta > 0 . \tag{7}$$

Obviously, $\tilde{E}_d$ is an asymptotically-decreasing function that approaches 0 when $\theta \to \infty$, see Figure 3 for an illustration.

Reformulating E_r. The reformulation of the E_r term is mathematically more complicated. Since $\hat{w}$ and $\hat{u}$ are independent of θ, when θ varies, their contributions to the integral can be considered as a scaling factor. Thus, they can be taken out of the integral and absorbed into a scaling parameter λ by forming a new parameter $\tilde{\lambda}$, making the new formulation of E_r data independent, which is our novel observation when simplifying the E_r term. In addition, by taking the two quantities out of the integral, we only leave $\mathbf{k}$ and G, which can be equivalently evaluated in the log-mapped Kolmogorov energy spectrum domain. Thus, E_r can be simplified and reformulated as:

$$\tilde{E}_r = \tilde{\psi}_r \int_0^1 k^2 (1 - \mathcal{G}(\theta))^2 \mathrm{d}k = \tilde{\psi}_r H(\theta), \tag{8}$$

where $\tilde{\psi}_r = (\int_0^1 k^2 (1 - \mathcal{G}(\theta_u))^2 \mathrm{d}k)^{-1}$ is for normalization. Inserting the specific form of $\mathcal{G}(\theta)$ into $H(\theta)$, we can obtain:

$$H(\theta) = \int_0^{k_1} k^2 [1 - e^{-\theta(k-k1)^2}]^2 \mathrm{d}k + \int_{k_2}^1 k^2 [1 - e^{-\theta(k-k_2)^2}]^2 \mathrm{d}k. \tag{9}$$

The two separate integrals can also be approximated by extending outside the range $[0,1]$ similarly as the integral approximation for E_d term, which leads to the following integral approximation:

$$\begin{aligned} H(\theta) \simeq\ & \textstyle\int_0^{k_1} k^2 \mathrm{d}k + \int_{k_2}^1 k^2 \mathrm{d}k \\ & - 2\left(\textstyle\int_{-\infty}^{k_1} k^2 e^{-\theta(k-k_1)^2} \mathrm{d}k + \int_{k_2}^{+\infty} k^2 e^{-\theta(k-k_2)^2} \mathrm{d}k \right) \\ & + \left(\textstyle\int_{-\infty}^{k_1} k^2 e^{-2\theta(k-k_1)^2} \mathrm{d}k + \int_{k_2}^{+\infty} k^2 e^{-2\theta(k-k_2)^2} \mathrm{d}k \right). \end{aligned} \tag{10}$$

Note that a general definite integral from calculus is:

$$\begin{aligned} \textstyle\int_a^b x^2 e^{-\theta(x-\alpha)^2} \mathrm{d}x =\ & \tfrac{1}{4\theta^{3/2}} (2\sqrt{\theta}(a+\alpha) e^{-\theta(a-\alpha)^2} \\ & - \sqrt{\pi}(2\alpha^2\theta + 1)\mathrm{erf}(\sqrt{\theta}(a-\alpha)) \\ & - 2\sqrt{\theta}(\alpha+b) e^{-\theta(b-\alpha)^2} \\ & + \sqrt{\pi}(2\alpha^2\theta + 1)\mathrm{erf}(\sqrt{\theta}(b-\alpha))) , \end{aligned} \tag{11}$$

where $\mathrm{erf}(\cdot)$ is the error function, which is still an integral. However, we will later find that such an integral can be canceled out. By

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

using such an integration result for Eq. 10, we further obtain:

$$\frac{\int_{-\infty}^{k_1} k^2 e^{-\theta(k-k_1)^2}\mathrm{d}k + \int_{k_2}^{+\infty} k^2 e^{-\theta(k-k_2)^2}\mathrm{d}k =}{\frac{1}{2\theta^{3/2}}[\sqrt{\pi}(\theta(k_1^2+k_2^2)+1) - 2\sqrt{\theta}(k_1-k_2)],} \tag{12}$$

and

$$\frac{\int_{-\infty}^{k_1} k^2 e^{-2\theta(k-k_1)^2}\mathrm{d}k + \int_{k_2}^{+\infty} k^2 e^{-2\theta(k-k_2)^2}\mathrm{d}k =}{\frac{1}{2^{5/2}\theta^{3/2}}[\sqrt{\pi}(2\theta(k_1^2+k_2^2)+1) - 2\sqrt{2\theta}(k_1-k_2)].} \tag{13}$$

We can see that the error function which is defined by an integral disappears, and we therefore obtain a purely analytical expression:

$$H(\theta) = \frac{\gamma}{3} - \frac{2(2^{3/2}-1)\sqrt{\pi}\alpha\theta + 3\sqrt{2}\beta\sqrt{\theta}+\eta}{2^{5/2}\theta^{3/2}}, \tag{14}$$

where $\alpha = k_1^2+k_2^2$, $\beta = 2(k_2-k_1)$; $\gamma = 1+k_1^3-k_2^3$; $\eta = (2^{5/2}-1)\sqrt{\pi}$; and $\tilde{\psi}_r = H(\theta_u)^{-1}$. From the formulation of $\tilde{E}_r$, it is seen that the variation of $\tilde{E}_r$ is consistent with the observation: when θ increases, the filter is narrowed and the ringing effect becomes more apparent as $\tilde{E}_r$ becomes larger, see Fig 3 for an illustration.

Analytical objective function. Lastly, we reformulate our objective function by combining $\tilde{E}_d$, $\tilde{E}_r$, and $\tilde{\lambda}$:

$$\tilde{E} = \tilde{E}_d^m + \tilde{\lambda}\tilde{E}_r^m = \left(\frac{\theta_l}{\theta}\right)^{\frac{m}{2}} + \tilde{\lambda}\left[\frac{H(\theta)}{H(\theta_u)}\right]^m. \tag{15}$$

Note that the value of θ_l and θ_u remain to be determined. Since $\tilde{E}$ is an approximation of E, θ_l should not be small; otherwise, the behavior of the objective function may be undesirable. As noted from the formulation of the filter, θ corresponds to the standard deviation of the Gaussian fall-off by $\theta = 1/\sigma^2$. To keep the filter sharpness, σ should not be large. Since we work in the normalized spectrum domain ($k \in [0,1]$), $\sigma = 0.2$ is sufficiently large, corresponding to $\theta = 25$. Thus, we assign $\theta_l = 25$. When the filter approaches an ideal BPF, θ would become very large. To determine θ_u, we use a relatively large value instead for θ and assign $\theta_u = 10^4$.

Originally, $\tilde{\lambda}$ is a user-tuned parameter for regularizing the ringing: if we increase $\tilde{\lambda}$, more nearby scales are taken into the decomposition to suppress the ringing. However, not all $\tilde{\lambda}$ values are acceptable. Hence, we need to determine appropriate $\tilde{\lambda}$ values, preferably in an automatic manner, which is described in the following.

Determination of $\tilde{\lambda}$. Experimentally, we found that to yield an optimizable problem, i.e., a minimizable $\tilde{E}$, $\tilde{\lambda}$ should be larger than a certain critical value. This forms the *optimizability constraint*, which makes $\tilde{E}$ almost flat beyond a certain θ as in [MLSF14]; see Figure 3. Such a constraint requires $\tilde{E}$ to have small gradients for all its values beyond the minimum of $\tilde{E}$, thus allowing us to automatically determine an appropriate $\tilde{\lambda}$.

Denoting the upper limit of θ as θ_u, and the derivatives of $\tilde{E}_d$ and $\tilde{E}_r$ at θ_u as $\tilde{E}_d'(\theta_u)$ and $\tilde{E}_r'(\theta_u)$, respectively, we compute $\tilde{\lambda}$ by solving the equation $\partial_\theta \tilde{E}(\theta_u) = \varepsilon$ with the identity $\tilde{E}_r^{m-1}(\theta_u) = 1$, which gives:

$$\tilde{\lambda} = \frac{\varepsilon}{m\tilde{E}_r'(\theta_u)} - \tilde{E}_d^{m-1}(\theta_u)\frac{\tilde{E}_d'(\theta_u)}{\tilde{E}_r'(\theta_u)}, \tag{16}$$

where the parameter ε is a fixed small gradient value chosen to be 5×10^{-8} in all our experiments. Note that such automatic calculation of $\tilde{\lambda}$ has already included the scaling introduced by taking $\hat{w}$ and $\hat{u}$ out of the integral. This is also another reason why we need an automatic procedure to compute $\tilde{\lambda}$.

3.1.5. Solving the Optimization and Decomposing the Flow

After computing $\tilde{\lambda}$, we employ Brent's algorithm [Bre02] to quickly search for an appropriate θ to minimize the objective function (Eq. 15). With the automatic method to calculate $\tilde{\lambda}$ (see Eq. 16), our optimization model is ensured to be unconditionally stable because the objective function is guaranteed to always attain the minimal value within the search range. In addition, the optimization model always converges effectively to the desired solution. We want to emphasize that due to our new formulation, the calculation of optimal θ is no longer data-dependent, and can be precomputed for all decomposed scales.

Figure 12 presents a comparison between our decomposition results and those from a perfect BPF (without any optimization) and from the curvelets method (we ignore the results from KoSCO since our results look almost the same as KoSCO's). It is apparent that the decomposition results from perfect BPF may introduce unexpected structures, which are inappropriate to use. Comparing with curvelets, our optimization also demonstrates closeness to the corresponding spectrum in the results, particularly for preserving structures appropriately for the corresponding scales.

To avoid potential ringing artifacts from the boundary reflection of non-periodic data sets, we create a mirror extension of the data to enforce periodicity in the computation, which is also required in other methods, e.g., curvelets, see Yang et al. [YP11]. Figure 4 shows one of our 2D flow decomposition results (see the supplementary video for a continuous version of it), and the following summarizes the whole scale decomposition procedure:

- **[Step 1]:** first, we define a scale-decomposition filter (Eq.1) in the log-mapped Kolmogorov energy spectrum domain, with a parameter θ to be determined.
- **[Step 2]:** then, we construct the objective function (Eq. 15) in the log-mapped Kolmogorov energy spectrum domain.
- **[Step 3]:** by calculating the derivatives of $\tilde{E}_d$ and $\tilde{E}_r$ at θ_u, we compute $\tilde{\lambda}$ according to Eq. 16.
- **[Step 4]:** given $\tilde{\lambda}$ and an initial θ ($\theta_0 = (\theta_l+\theta_u)/2$), we employ Brent's algorithm to efficiently find the optimal filter parameter θ_m according to the objective function.
- **[Step 5]:** since the filter is formulated to be data independent, we can pre-compute the filters for all the scales we need for scale decomposition by giving different k_1 and k_2 in the log-mapped Kolmogorov energy spectrum and solving the optimization.
- **[Step 6]:** lastly, we convert the optimized filter back to Fourier domain, multiply with the original data, and perform an inverse Fourier transform to obtain the corresponding scale decomposition results.

3.2. Continuous-scale Visualization

After the flow decomposition, turbulence structures hidden in different scales of the input flow fields can be revealed by visualizing the magnitude of the decomposed flow. To this end, we employ

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

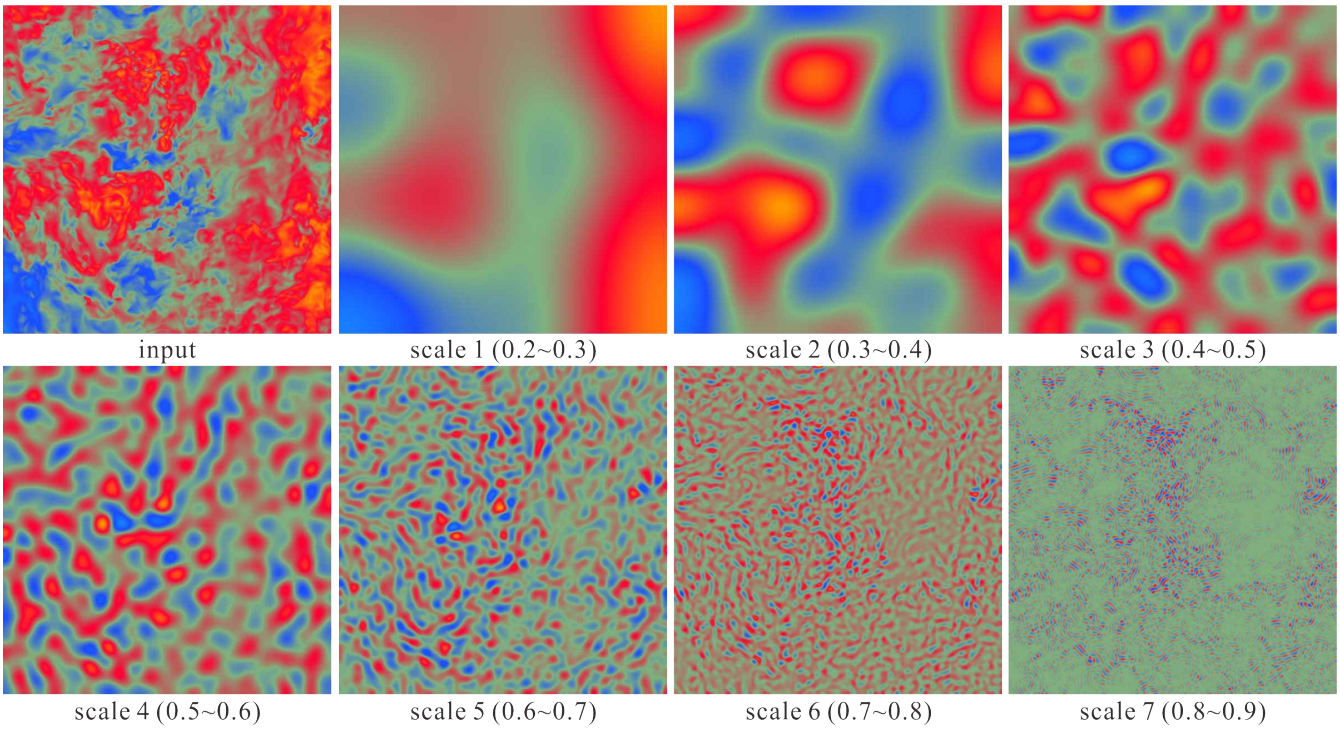

Figure 4: *Continuous-scale decomposition on a* 512×512 *isotropic DNS turbulent flow data. Red and blue colors indicate high and low velocities, respectively.* Please refer to the supplementary video for a continuous version of this decomposition result.

the isosurface visualization to produce 3D renderings of the decomposed flow field, since it is commonly adopted by the domain scientists. Following the convention in the turbulence community [BMP08], the *velocity magnitude* of the decomposed flow field is assumed to conform to a Gaussian distribution since turbulent flows are chaotic by nature. Hence, we follow [BMP08] to use $\mu + 1.5\sigma$ as the isovalue to produce the isosurface meshes by a high-resolution Marching cubes algorithm [LC87], where μ and σ are the mean and standard deviation of the decomposed velocity field, respectively. Such a selection of isovalue was found in previous work for a clear and rich representation of the flow structures.

However, unlike the conventional isosurface rendering, which shades the scale isosurfaces with a constant color, in this work, we propose a new style of visualization, which combines isosurface extraction from the scale-decomposed flow fields with surface texturing from the color map of the input flow velocity, e.g., see Figure 5, where the color indicates the magnitude of the flow velocity. Traditional isosurface can only show spatial structures in a particular scale. By texturing the surface with color-mapping from the corresponding velocity field in the domain, we can better reveal and relate the distributions of turbulence scale structures with respect to the input flow fields. This can lead to more intuitive visual analysis, similar to the λ_2 flow visualization [SLÖ*14].

In our decomposition framework, we can flexibly decompose an input flow field into continuous scales, and produce *continuous-scale visualization* to smoothly reveal the variation of length scales as distributed over the input flow field. This is particularly useful for showing fractal-like turbulence structures. See Figures 5, 8, and 10 for our continuous-scale visualization results. It is worth noting that the conventional flow decomposition methods with wavelets and curvelets can only produce *discrete* rather than *continuous* scale decompositions. Continuous visualization of length-scale structures is highly desirable for scientific study of turbulent flows, especially in transitional regions of the boundary layer flow [Jim13]. However, it has not been explored and presented in any of the existing flow visualization work we are aware of. Note also that since we can only show static images of particular decomposed scales in the paper, *readers can refer to the supplementary video for the continuous versions of these decomposition results.*

4. Results and Discussions

We implement our scale decomposition method and its related visualization on a workstation with Intel Core (TM) i7-3930K CPU@3.20GHz, 28GB RAM and 2TB hard drive. Since we base our optimization framework on an analytic objective, it takes only around 0.02 sec. (experimentally over different scales) to optimize the filter shape for a given scale. Note that the filter optimization is independent of the data sets, which can be verified by referring to our optimization formulation, where the filter parameters are actually independent of the data values. After we obtain the filter shape for a given scale, we then extract the turbulence structures related to the scale by a filtering in the Fourier space, whose computational time depends on the size (resolution) of the data set. For a 2D data set of resolution 512^2, our method takes around 0.3 seconds for the filtering, while for a 3D isotropic turbulence data set of resolution 512^3 and a 3D boundary layer turbulence data set of resolution

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

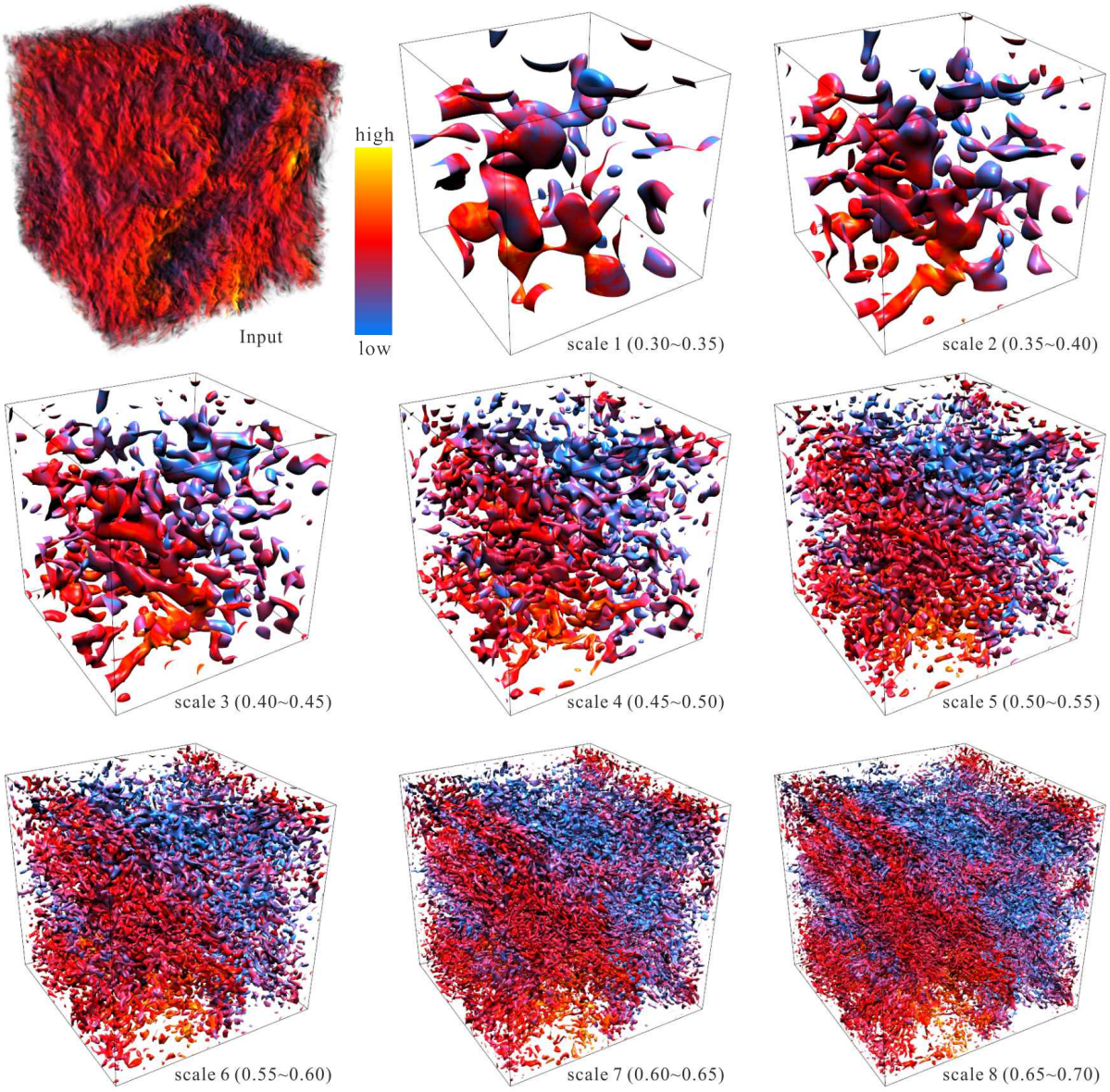

Figure 5: *Continuous-scale visualization of turbulence structures in an isotropic turbulent flow. The color indicates the magnitude of the flow velocity, with red and yellow colors indicating large velocities and blue color indicating small velocities.*

$1000 \times 257 \times 1024$ (to be shown later), our method takes around 5 and 2 minutes, respectively, for the filtering. Compared with [MLS-F14], our method accelerates the computation by around five times with compelling decomposition results.

4.1. Visualization Results

In the following, we present case studies of visualizing a variety of turbulent flows using our method. Note that there are two classes of turbulent flows in general: isotropic and sheared flows [Tri88]. For isotropic flows, we consider a typical turbulent flow by an initial random force in a periodic cube, whereas for sheared turbulent flows, we consider a boundary layer case on a long flat plate, where the no-slip condition is applied at the plate boundary. They are both typical representatives of the two basic turbulent flow classes. Thus, we use their DNS flow data sets, particularly with different parameters such as varying Reynolds numbers in the boundary layer flows, for the visual examination of the scale structures in the turbulent flow fields. Note also that the analysis below is done with the involvement of domain experts in turbulence research.

4.1.1. Isotropic Turbulence

The first data set we experimented with is an isotropic DNS turbulence data we obtained from John Hopkins University (JHU) tur-

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

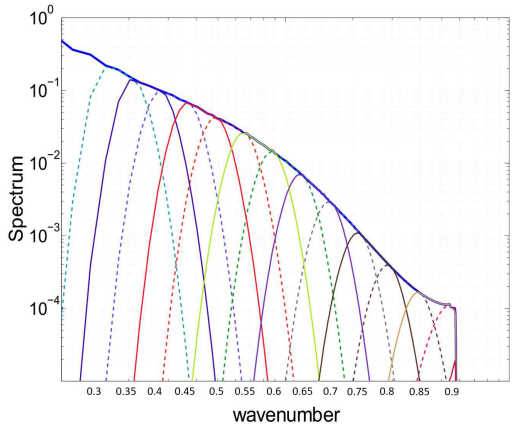

Figure 6: *Fourteen selected representative filters that are roughly evenly distributed in the Kolmogorov energy spectrum; the first eight filters starting from wavenumber* 0.3 *were used to produce the eight visualizations (scales 1-8) shown in Figure 5. Note that the solid and dotted lines indicate even-numbered and odd-numbered scales, respectively.*

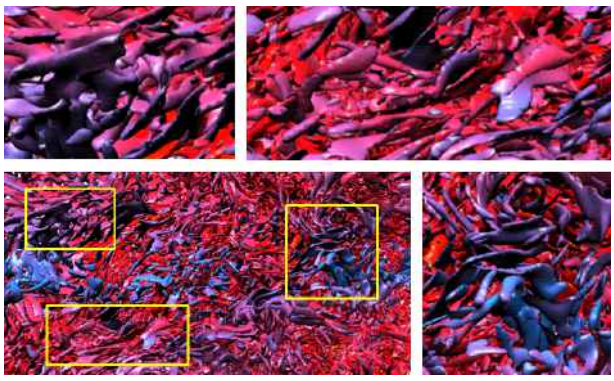

Figure 7: *A zoom-in view to show the smaller-scale dissipative range: scale [0.80,0.85] in the Kolmogorov energy spectrum. The yellow boxes above highlight some of the sheet-like structures in this smaller-scale isotropic turbulence.*

bulence database cluster [LPW*08] at $Re_\lambda = 433$. The simulation was done in a cube with a grid resolution of 1024^3.

Figure 5 shows our visualization results. Here, we use a bandwidth of size 0.05 (i.e., $\delta k = k_2 - k_1 = 0.05$ in Figure 2) in the Kolmogorov energy spectrum, smoothly shifting the scales like a sliding window, and optimize the related filters to produce the continuous-scale visualization. Again, since the figure can only show certain image instances in the continuous-scale visualization, Figure 5 shows only the visualizations of the eight larger scales as the representatives. See Figure 6 for the corresponding optimized filters in the Kolmogorov energy spectrum, where we pick 14 (roughly evenly distributed) scales from the continuous filter space. Note also that if we use curvelets to decompose this data set for visualization, only six discrete scales can be obtained due to the dyadic (powers-of-two) scale limit, while by using our decomposition framework, we can flexibly specify a scale and explore turbulence structures continuously in the Kolmogorov energy spectrum.

The first three scales with larger-scale structures (i.e., scales 1 to 3 in Figure 5) are in the forcing range of the Kolmogorov energy spectrum. They contain most of the turbulent energy and form the larger-scale structures. In these scales, thick tubes or blob-like structures can be observed. The rest of the scales with smaller-scale structures (i.e., scales 4 to 8 in Figure 5) correspond to the inertial range, where thinner tube-like structures dominate. The tubes become even thinner and more stretched as we continuously move to smaller scales, indicating a turbulent stretching process. This dissipative range of decomposed scales are not presented in Figure 5 because the related structures are too small to be recognizable if we show their visualizations in the same size as the scales 1-8 in Figure 5. Hence, we use a separate zoom-in figure, i.e., Figure 7, to show a small-scale example in the dissipative range, where the camera is located inside the turbulent flow region with the same view direction as in Figure 5, and the zoom-in view-port is only around $1/20$ of the whole region projected onto the imaging plane.

Our visualization results are consistent with previous work on multi-scale (discrete and fixed number of scales) turbulence analysis. For example, Moreno et al. [BMP08] also observed blob-like structures in larger-scale energy-containing range. This is similar to our decomposition results in larger scale, i.e., scales 1 to 3 in Figure 5. Moreover, they also observed sheet-like structures in the dissipative range, and this can also be seen in our decomposition results, see the boxed regions in Figure 7 for examples of sheet-like structures in isotropic turbulence.

4.1.2. Boundary Layer Turbulence

A classic example of anisotropic turbulence is the turbulent boundary layer flow over a flat plate. Here, we use two DNS boundary layer turbulence data sets we obtained from Royal Institute of Technology (KTH) Sweden at $Re_\tau = 1000$ and $Re_\tau = 4000$ [SÖ10] to explore the capability of our method for anisotropic turbulence. The simulations that produced these data sets were done in a very high resolution rectangular domain, where we cropped a $1000 \times 257 \times 1024$ region for visualization and analysis. Note that unlike isotropic turbulence, the grid along the wall normal direction (+y) is stretched to account for the wall effects; see Figure 8 (top left).

Decomposing turbulence scale structures in boundary layer flows requires special treatment, since the scales for a turbulent boundary layer flow along the wall normal direction is not well defined in terms of the Kolmogorov energy spectrum due to the data non-periodicity and the stretching of the simulation coordinates along wall normal direction, where the Fourier transform is not applicable. Hence, we can only decompose the flow field over the X-Z plane, i.e., the streamwise and spanwise directions, see again Figure 8 (top left). As a result, we perform our decomposition 2D slice by 2D slice along the Y axis, and then stack the 2D decomposed results to form the overall flow decomposition. Since we do not discard any point from the data, the data resolution remains unchanged after the decomposition. Figures 8 and 10 show our continuous-scale decomposition results of the turbulent boundary

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

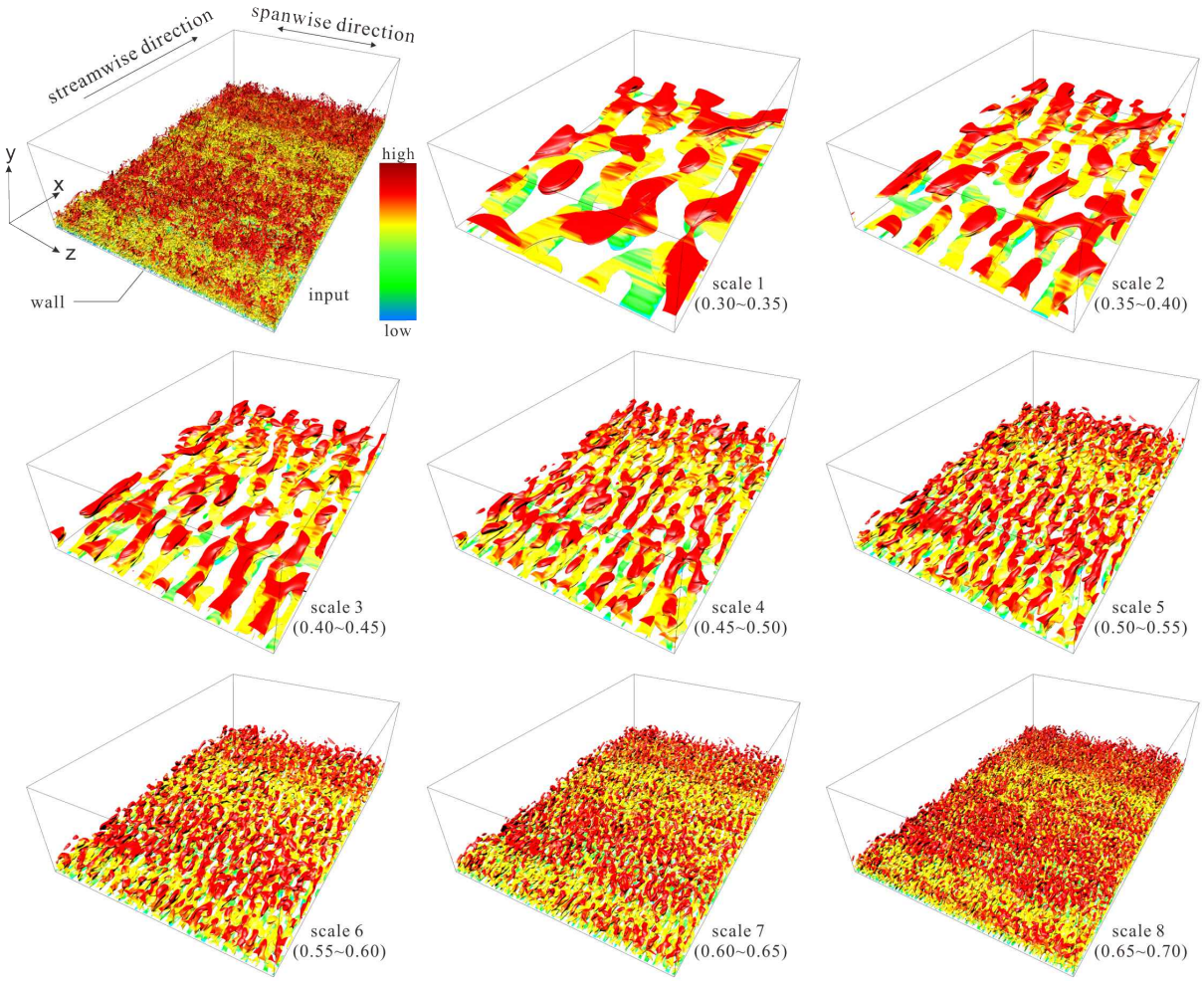

Figure 8: *Continuous-scale visualization of boundary layer turbulence at low Reynolds number 1000. The image on top left shows the input flow velocity field, which is visualized using the standard λ_2 isosurface visualization method. The color indicates the magnitude of flow velocity.*

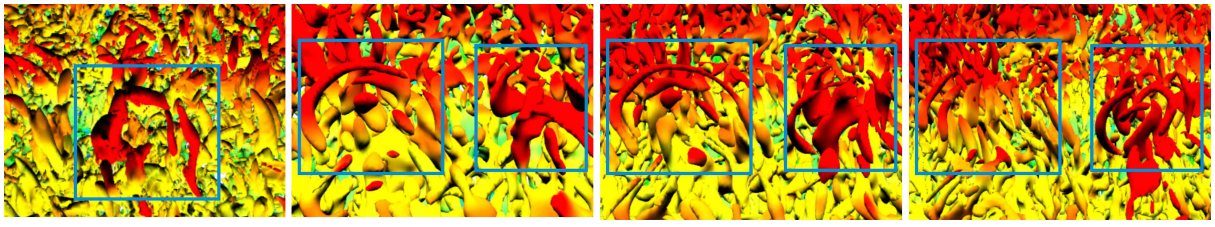

Figure 9: *Hairpin structures observed in different visualizations: (a) the hairpin structure in standard λ_2 visualization of the original flow; and (b) to (d) hairpin structures observed in our visualizations, as continuously distributed over nearby scales. As we go across scales, we may continuously visualize the formation, evolution, and splitting of these structures. Note that the hairpin structures in λ_2 may not exactly match the continuous-scale visualizations.*

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

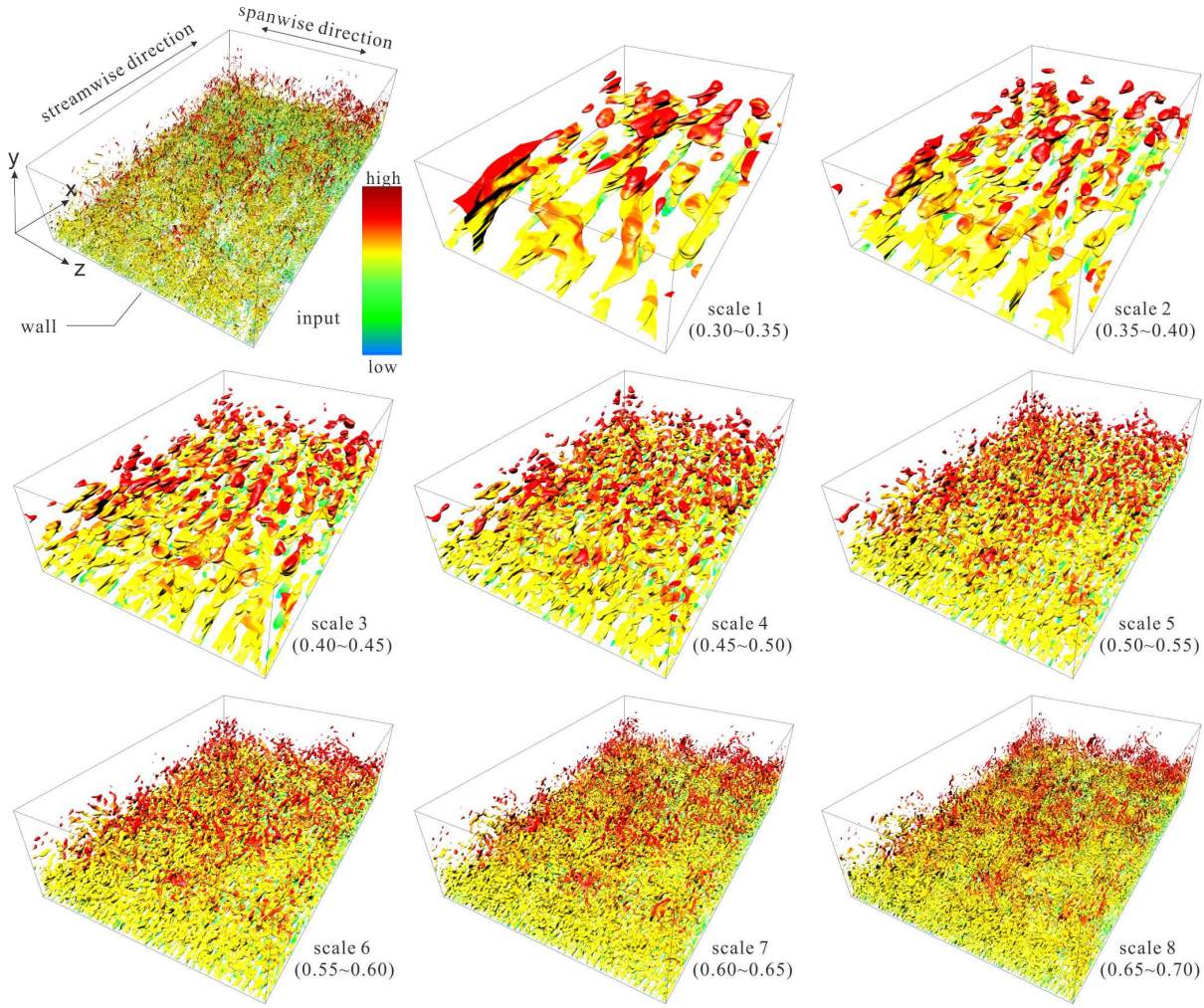

Figure 10: *Continuous-scale visualization of boundary layer turbulence at high Reynolds number 4000. Again, the image on top left shows the input flow velocity field, which is visualized using standard λ_2 isosurface visualization method. The color indicates the magnitude of flow velocity.*

layer flows at low and high Reynolds numbers. Like isotropic turbulence, we present mainly the larger scales in these figures.

Low Reynolds number boundary layer flow. If we zoom into the visualizations in Figure 8, we can find some important Ω-shaped structures known as the hairpin structures, see Figure 9. These structures were initially observed in the isosurface visualization of λ_2 features, see Figure 9(a), but from our visualization results, see Figure 9(b-d), we can see that such structures are also captured, but at relatively smaller scales, indicating that hairpin structures are usually formed with small-scale structures. Across scales, we can see their formation, evolution, and splitting. However, the location of hairpin structures in our continuous-scale visualization does not necessarily match with those from the overall visualization of λ_2 features since they exist in different domains.

Moreover, unlike λ_2 hairpin structures, which cannot tell what scale the structures are associated with in the Kolmogorov energy spectrum, our method can pinpoint the scales for the structures in a continuous manner, see again Figure 9. Hence, we can enable more precise length-scale analysis of hairpin structures, which λ_2 visualization cannot offer. Note also that such analysis has not been achieved in any existing turbulence research.

High Reynolds number boundary layer flow. Figure 10 presents isosurface visualizations of some larger scales selected from our continuous-scale visualization. From the visualizations, we can find that most larger-scale structures are further away from the wall (y=0), while smaller-scale structures occur relatively closer to the wall. These visualizations are consistent with the physical property of boundary layer turbulent flows, which are known to consist of two layers: an outer layer, which tends to have larger length scales, and an inner layer, which tends to possess smaller length scales. Furthermore, it is interesting to note that the Ω-shaped structures are not visible for this case, which is consistent with the all but vanishing hairpin structures observable with λ_2 [SLÖ*14], thus indicating that these structures are associated with the low-Reynolds number transitional boundary layer only.

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

Scale	k_1	k_2	θ_m	$\sigma_m = \sqrt{1/(2\theta_m)}$
1	0.10	0.15	770.17	0.02547
2	0.15	0.20	789.41	0.02517
3	0.20	0.25	803.81	0.02494
4	0.25	0.30	818.87	0.02471
5	0.30	0.35	835.51	0.02446
6	0.35	0.40	853.09	0.02421
7	0.40	0.45	872.61	0.02397
8	0.45	0.50	893.75	0.02365
9	0.50	0.55	915.73	0.02337
10	0.55	0.60	942.51	0.02303
11	0.60	0.65	968.29	0.02272
12	0.65	0.70	996.53	0.02239
13	0.70	0.75	1023.12	0.02211
14	0.75	0.80	1054.54	0.02177
15	0.80	0.85	1088.58	0.02143
16	0.85	0.90	1122.53	0.02111
17	0.90	0.95	1158.62	0.02077
18	0.95	1.00	1195.77	0.02045

Table 1: *Filter parameters estimated for different scales in the log-mapped Kolmogorov energy spectrum; σ_m is the standard deviation of Gaussian to measure the sharpness of the filter fall-off regions.*

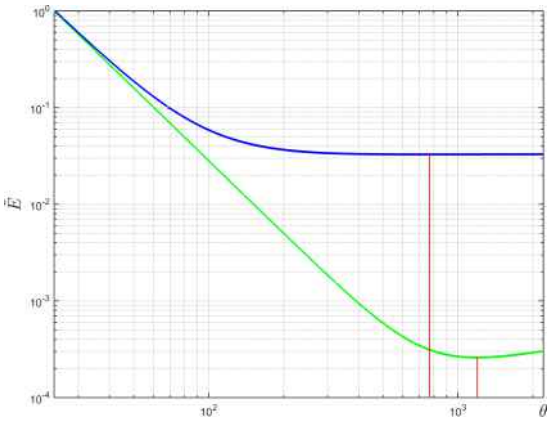

Figure 11: *The shape of the objective functions and the optimal values of θ (marked by red lines) for decomposing scales 1 (blue curve) & 18 (green curve) in Table 1.*

4.2. Filter Parameter

Our optimization process starts with a given scale, i.e., k_1 and k_2, and optimizes the related filter parameter, see Figure 2. Table 1 shows the estimated filter parameters for different given scales, where $\sigma_m = \sqrt{1/(2\theta_m)}$ is the standard deviation of the Gaussian function, indicating the sharpness of the fall-off regions; a larger θ_m indicates a sharper fall-off.

From Table 1, we can see that as we move from larger to smaller scales (top to bottom in the table), θ_m increases; hence, it shows an increase in the sharpness in the fall-off region defined in the log-space of the Kolmogorov energy spectrum. However, it does not mean that there is an increase in the sharpness of the fall-off region in the Fourier space since there is a log-transformation, and the filters for smaller scales are more compressed. In fact, as we go to smaller scales in the Fourier domain, the sharpness of the fall-off region decreases. This indicates stronger ringing artifacts at smaller scales, thus requiring more nearby scales to suppress them.

To demonstrate how the optimal filter parameter (θ_m) affects the optimization results, we present Figure 11, which shows the objective function values of the largest (blue curve) and the smallest (green curve) scales in Table 1, with the red lines indicating the corresponding locations of θ_m in the spectrum. It is noted that smaller scales have larger decreasing regions in the objective function, thus leading to a larger value of θ_m.

4.3. Ringing Artifacts

In our flow decomposition framework, we aim to maintain a sharp filter shape while minimizing the ringing artifacts. To show that our method can still effectively minimize the ringing artifacts while maintaining a sharp filter, Figure 12 compares the results from our method with the results from the curvelets method [BMP08], as well as the results from the perfect band-pass filtering in Fourier space. In the first row of Fig 12, we can see that since the Fourier basis has only global support, a perfect band-pass filter may discard necessary basis functions, and thereby produce oscillatory ringing structures in the decomposed results. On the other hand, in the second row of Figure 12, we can see that although the curvelets method can suppress the ringing problem with local-support basis, certain amount of ringing artifacts still remain.

In Figure 12, we generate our decomposition results by constructing an optimized filter shape from each scale used in the curvelets method. Although ringing artifacts cannot be completely avoided especially for smaller scales, our method can still effectively minimize the ringing artifacts with a quality which is comparable to and sometimes better than that of the curvelets method. Note that our method does not generate strong intermittency effects: the laminar region will still remain laminar, as can be seen from the plane regions around the fractal in first column of Figure 12. In this case, curvelets method still produces some ringing while our method performs better without obvious ringing.

4.4. Kolmogorov Energy Spectrum Space Filtering

Our filters, which are constructed in the Kolmogorov energy spectrum space, are essentially filters in the Fourier space, but with non-linear logarithmic mapping. Thus, the decomposition is not purely a linear decomposition. In addition, the scale decomposition results do not necessarily satisfy the underlying Navier-Stokes equations. Rather, they represent the structure of the solution at particular scales for better visualization and analysis to understand the turbulence scale characteristics.

Note that our scale decomposition is not a local decomposition, and the proposed method does not aim to detect local scale phenomena. On the contrary, it extracts global scale structures as in any previous work, like [BMP08] and [MLSF14], which is the reason that filtering in Kolmogorov energy spectrum space can be adopted in this paper. Wavelets/curvelets-based approaches only utilize the

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

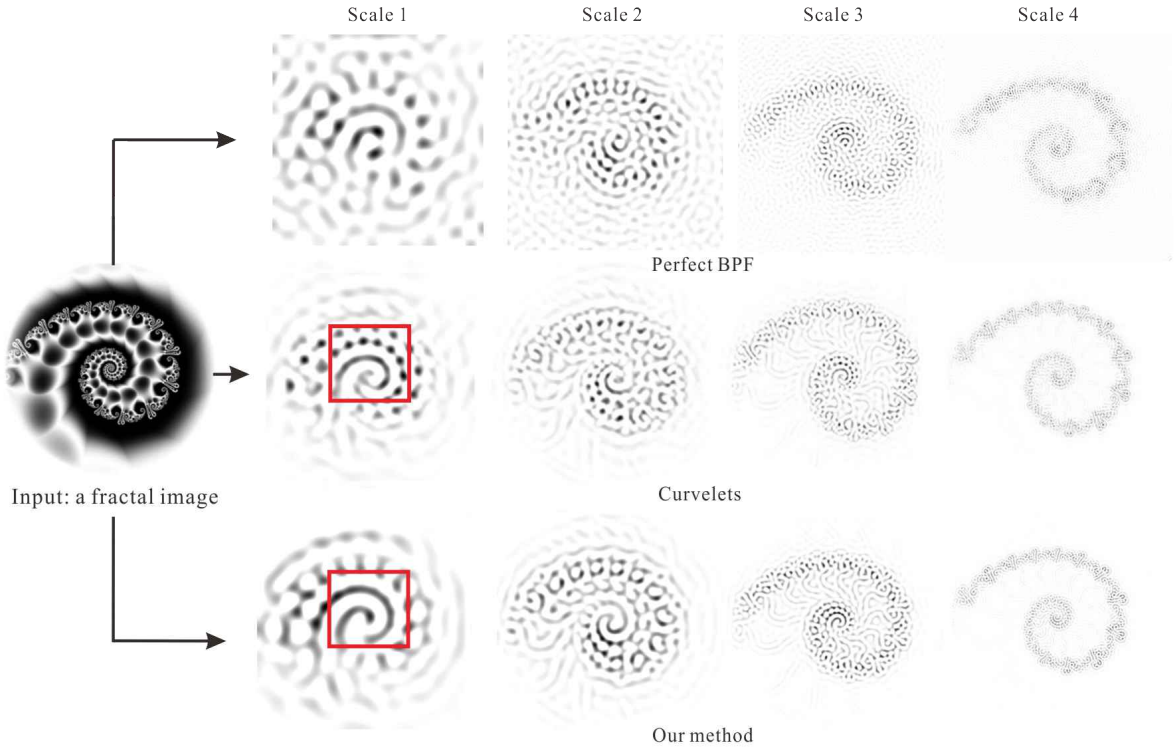

Figure 12: *We compare the scale decomposition quality of our method against the curvelets method [BMP08] and perfect BPF using a fractal image (Julia fractal image), which exhibits spatial structures of varying scales; here, we follow Bermejo-Moreno and Pullin, who conducted experiments on curvelets-based method by analyzing multi-scale features in this fractal image. Visual comparison above shows that our results are comparable to the curvelets results. See the intermittencies illustrated in the red box of the curvelets result. Our method can, however, mostly avoid them with better structure preservation.*

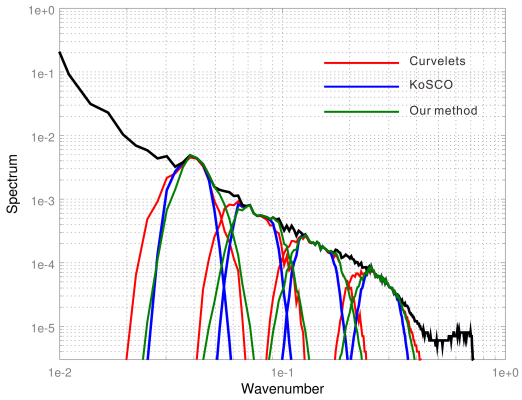

Figure 13: *Comparison with curvelets-based decomposition method [BMP08] and KoSCO [MLSF14];. It is noted that although we employ completely different approach, our result is similar to curvelets-based method and KoSCO especially for smaller scales, indicating ring minimization property of our method.*

multi-scale nature of wavelets/curvelets for scale decomposition, as well as their ringing minimization feature. Our comparisons in Figures 12 & 13 verify that our decomposition method produces very similar results as the curvelets-based approach.

Remark: please refer to the supplemental video for animated results and additional comparisons. In addition, we will prepare a project website and release the code for decomposing and visualizing turbulent flow scales.

4.5. Evaluation

4.5.1. Energy Spectrum Distribution

To verify our decomposition results, we compare the energy spectrum distribution of our results to that from curvelets, which have been widely used in turbulence research community, and that from our previous work [MLSF14]. To prepare for this experiment, we use the fractal image (see Figure 12 (middle left)) from Bermejo-Moreno and Pullin [BMP08], who also conducted experiments with curvelets decomposition methods and considered the fractal image since it exhibits multi-scale features. Moreover, we estimate the extent of the scales in the energy spectrum from the curvelets decomposition result, and then use them as the input parameters in our method to obtain our scale filters.

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

Figure 13 presents the corresponding energy spectrum distributions for curvelets results [BMP08], results from [MLSF14] and ours. From Figure 12, we observed previously that our decomposition results are similar to those from curvelets. Furthermore, from Figure 13, we can also see that the three spectrums are similar in shape, but the spectrums for curvelets have more fluctuations and jerkiness, which may lead to artificial intermittency structures, e.g., see the red box in Figure 12. On the contrary, due to our optimization formulation, our method can better preserve the structures (see again Figure 12). Furthermore, our method allows the specification of scales continuously spreading over the Kolmogorov energy spectrum (for continuous-scale visualization), while curvelets methods can only provide a maximum of six scales (for 512×512 images) due to dyadic scaling.

4.5.2. Feedback from a Domain Scientist

We invited a domain expert in turbulence research from Royal Institute of Technology (KTH) Sweden to help evaluate our results. In his expert review, he stated that "the inertial range located in spectral space between the integral and viscous scales is dominated by a continuous distribution of energy in scale space. Therefore, any decomposition of turbulence data sets needs to take into account this range of scales, either by filtering in physical space (e.g. based on volume or life time), or in spectral space as proposed in the present article. Thus, I can clearly confirm the motivating statements by the authors in that a continuous-scale decomposition is relevant." He further stated that "the turbulence cascade is transporting energy through from large to small scales in scale space. This process is very complex on an individual structure level, and may involve merging and splitting of structures in a chaotic way, which means that individual structures move continuously through scale space. Only by being able to fine-tune the relevant scales in a specific decomposition, one is able to visualize and eventually track an individual element of the cascade. It is exactly this insight that will lead to better understanding and modeling possibilities of the complex phenomenon of turbulence." Finally, he concluded by saying "I can confirm that the premise the authors started with, i.e. the importance of a continuous-scale decomposition, is indeed relevant to turbulence. I can definitely see the potential of the method. This type of continuous-scale visualization of structures might prove relevance to improving structure-based turbulence models."

5. Conclusion

In this paper, we present a novel optimization-based technique to decompose a turbulent flow field into scale components in the Kolmogorov energy spectrum. The decomposed scales can be continuously specified, and our method can enable us to deliver *continuous-scale decomposition* and *continuous-scale visualization* of the decomposed turbulence structures.

Our method is derived from an analytical optimization objective function, which can be solved with particularly high efficiency to produce an optimal filter shape that maintains a sharp filter shape while reducing the amount of ringing. By this, we can produce high-quality scale decomposition that is comparable (and sometimes superior) to the state-of-the-art methods. Moreover, our computing time is much shorter, making our method suitable for processing large volume of data sets with higher resolutions and dimensions, e.g., time-varying flow data sets. We also applied our method on DNS data sets, including isotropic and boundary layer turbulent flows; the results show that our visualization can unveil hidden turbulence structures such as blob-like, sheet-like and hairpin structures in different scales in a continuous manner.

In the future, we would like to explore time-varying DNS turbulence data sets in order to track turbulence scale structures for further analysis. This is particularly useful for turbulent boundary layer flows to understand the dynamics of hairpin structures: their birth, evolution, merging, and splitting. Since our method is highly efficient for scale decomposition, it can help to enable structure tracking in a continuous-scale manner.

Acknowledgement

The authors would like to thank all reviewers for their constructive comments, as well as Professor Philipp Schlatter for discussing the turbulence scale decomposition and visualization. This work is supported by the National Natural Science Foundation of China (NSFC) - Outstanding Youth Foundation (Grant No. 61502305), the startup funding of ShanghaiTech University, as well as the Program of Shanghai Subject Chief Scientist (A type) (No.15XD1502900).

References

[ABAPØCE07] ANDERS H., B. ANDERS PETTERSSON R., ØYVIND A., CARL ERIK W.: Visualization of vorticity and vortices in wall-bounded turbulent flows. *IEEE Transactions on Visualization and Computer Graphics 13*, 5 (2007), 1055–1067. 3

[AØA*04] ANDERS H., ØYVIND A., ATLE O., B. ANDERS PETTERSSON R., JOE W., TROND G.: Visualization of the energy-containing turbulent scales. In *2004 IEEE Symposium on Volume Visualization and Graphics* (2004), pp. 103–109. 3

[BMP08] BERMEJO-MORENO I., PULLIN D. I.: On the non-local geometry of turbulence. *Journal of Fluid Mechanics 603* (4 2008), 101–135. 1, 2, 5, 7, 9, 12, 13, 14

[Bre02] BRENT R. P.: *Algorithms for minimization without derivatives*. Dover Publications, 2002. 6

[CD99] CANDES E., DONOHO D.: *Curvelets: A Surprisingly Effective Nonadaptive Representation for Objects with Edges*. Technical report (Stanford University. Dept. of Statistics). Department of Statistics, Stanford University, 1999. 2

[CDDY06] CANDÈS E., DEMANET L., DONOHO D., YING L.: Fast discrete curvelet transforms. *Multiscale Modeling & Simulation 5*, 3 (2006), 861–899. 2

[DX97] DEBORAH S., XIN W.: Tracking and visualizing turbulent 3D features. *IEEE Transactions on Visualization and Computer Graphics 3*, 2 (1997), 129–141. 3

[Far92] FARGE M.: Wavelet transforms and their applications to turbulence. *Annual Review of Fluid Mechanics 24*, 1 (1992), 395–458. 1, 2

[GCS*12] GAITHER K., CHILDS H., SCHULZ K. W., HARRISON C., BARTH W., DONZIS D., YEUNG P.-K.: Visual analytics for finding critical structures in massive time-varying turbulent-flow simulations. *IEEE Computer Graphics and Applications 32*, 4 (2012), 34–45. 3

[GGTH07] GARTH C., GERHARDT F., TRICOCHE X., HAGEN H.: Efficient computation and visualization of coherent structures in fluid flow applications. *IEEE Transactions on Visualization and Computer Graphics, 13*, 6 (2007), 1464–1471. 3

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

[Gib99] GIBBS J. W.: Fourier's series. *Nature 59*, 606 (1899). 1

[JCG08] JOHNSON G. P., CALO V. M., GAITHER K. P.: Interactive visualization and analysis of transitional flow. *IEEE Transactions on Visualization and Computer Graphics 14*, 6 (2008), 1420–1427. 2

[JH95] JEONG J., HUSSAIN F.: On the identification of a vortex. *Journal of Fluid Mechanics 285* (1 1995), 69–94. 3

[Jim13] JIMÉNEZ J.: Near-wall turbulence. *Physics of Fluids 25*, 10 (Oct. 2013), 101302. doi:10.1063/1.4824988. 1, 7

[Kol41] KOLMOGOROV A. N.: The local structure of turbulence in incompressible viscous fluids at very large reynolds numbers. *Dokl. Nauk. SSSR 30* (1941), 301–305. 1

[Kol62] KOLMOGOROV A. N.: A refinement of previous hypotheses concerning the structure of turbulence in a viscous incompressible fluid at high reynolds number. *Journal of Fluid Mechanics 13* (4 1962), 82–85. 1

[KWDG11] KOEHLER C., WISCHGOLL T., DONG H., GASTON Z.: Vortex visualization in ultra low Reynolds number insect flight. *IEEE Transactions on Visualization and Computer Graphics, 17*, 12 (2011), 2071–2079. 2

[LC87] LORENSEN W. E., CLINE H. E.: Marching cubes: A high resolution 3D surface construction algorithm. *SIGGRAPH 1987 21*, 4 (Aug. 1987), 163–169. 7

[LPW*08] LI Y., PERLMAN E., WAN M., YANG Y., BURNS R., MENEVEAU C., BURNS R., CHEN S., SZALAY A., EYINK G.: A public turbulence database cluster and applications to study lagrangian evolution of velocity increments in turbulence. *Journal of Turbulence 9*, 31 (2008), 1–29. 9

[LSD12] LEUNG T., SWAMINATHAN N., DAVIDSON P. A.: Geometry and interaction of structures in homogeneous isotropic turbulence. *Journal of Fluid Mechanics 710* (10 2012), 453–481. 2

[MHVD09] MA J., HUSSAINI M. Y., VASILYEV O. V., DIMET F. L.: Multiscale geometric analysis of turbulence by curvelets. *Physics of Fluids 21*, 7 (2009), 075104. 1

[MLSF14] MISHRA M., LIU X., SKOTE M., FU C.-W.: Kolmogorov spectrum consistent optimization for multi-scale flow decomposition. *Physics of Fluids 26*, 5 (2014), 055106. 1, 2, 3, 4, 5, 6, 8, 12, 13, 14

[Pin02] PINSKY M. A.: *Introduction to Fourier Analysis and Wavelets*. American Mathematical Society, 2002. 5

[Pop01] POPE S. B.: *Turbulent Flows*. Cambridge University Press, 2001. 1, 3

[PTA*11] POBITZER A., TUTKUN M., ANDREASSEN O., FUCHS R., PEIKERT R., HAUSER H.: Energy-scale aware feature extraction for flow visualization. In *Proceedings of the 13th Eurographics / IEEE - VGTC conference on Visualization* (2011), EuroVis'11, pp. 771–780. 3

[RPP11] RAIESI H., PIOMELLI U., POLLARD A.: Evaluation of turbulence models using direct numerical and large-eddy simulation data. *Journal of Fluids Engineering 133*, 2 (2011). 2

[SBV*11] SCHAFHITZEL T., BAYSAL K., VAARANIEMI M., RIST U., WEISKOPF D.: Visualizing the evolution and interaction of vortices and shear layers in time-dependent 3D flow. *IEEE Transactions on Visualization and Computer Graphics 17*, 4 (2011), 412–425. 2, 3

[SDT06] SAGAUT P., DECK S., TERRACOL M.: *Multiscale And Multiresolution Approaches in Turbulence*. Imperial College Press, 2006. 1

[Sko13] SKOTE M.: Comparison between spatial and temporal wall oscillations in turbulent boundary layer flows. *Journal of Fluid Mechanics 730* (2013), 273–294. 2

[Sko14] SKOTE M.: Scaling of the velocity profile in strongly drag reduced turbulent flows over an oscillating wall. *International Journal of Heat and Fluid Flow 50* (2014), 352 – 358. doi:http://dx.doi.org/10.1016/j.ijheatfluidflow.2014.09.006. 2

[SLÖ*14] SCHLATTER P., LI Q., ÖRLÜ R., HUSSAIN F., HENNINGSON D.: On the near-wall vortical structures at moderate Reynolds numbers. *European Journal of Mechanics - B/Fluids 48* (2014), 75–93. 7, 11

[SÖ10] SCHLATTER P., ÖRLÜ R.: Assessment of direct numerical simulation data of turbulent boundary layers. *Journal of Fluid Mechanics 659* (2010), 116–126. 9

[SOL*13] SHAFII S., OBERMAIER H., LINN R., KOO E., HLAWITSCHKA M., GARTH C., HAMANN B., JOY K.: Visualization and analysis of vortex-turbine intersections in wind farms. *IEEE Transactions on Visualization and Computer Graphics, 19*, 9 (2013), 1579–1591. 2

[Spa15] SPALART P. R.: Philosophies and fallacies in turbulence modeling. *Progress in Aerospace Sciences 74* (Apr. 2015), 1–15. doi:10.1016/j.paerosci.2014.12.004. 1

[Str08] STROCCHI F.: *An Introduction to the Mathematical Structure of Quantum Mechanics: A Short Course for Mathematicians*. World Scientific, 2008. 4

[SUT05] SIMON S., ULRICH R., THOMAS E.: Opening the can of worms: an exploration tool for vortical flows. In *Visualization, 2005. VIS 05. IEEE* (2005), pp. 463–470. 2, 3

[SVG*08] SCHAFHITZEL T., VOLLRATH J. E., GOIS J. P., WEISKOPF D., CASTELO A., ERTL T.: Topology-preserving λ_2-based vortex core line detection for flow visualization. In *Proceedings of the 10th Joint Eurographics / IEEE - VGTC Conference on Visualization* (2008), EuroVis'08, Eurographics Association, pp. 1023–1030. 3

[TBR*12] TREIB M., BÜRGER K., REICHL F., MENEVEAU C., SZALAY A., WESTERMANN R.: Turbulence visualization at the terascale on desktop PCs. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2169–2177. 2

[Tri88] TRITTON D.: *Physical Fluid Dynamics*. Oxford Science Publications. Oxford University Press, 1988. 8

[vdB04] VAN DEN BERG J. C.: *Wavelets in Physics*. Cambridge University Press, 2004. 1

[WM09] WU X., MOIN P.: Direct numerical simulation of turbulence in a nominally zero-pressure-gradient flat-plate boundary layer. *Journal of Fluid Mechanics 630* (6 2009), 5–41. 2

[WPB*11] WILLIAMS S., PETERSEN M., BREMER P.-T., HECHT M., PASCUCCI V., AHRENS J., HLAWITSCHKA M., HAMANN B.: Adaptive extraction and quantification of geophysical vortices. *IEEE Transactions on Visualization and Computer Graphics, 17*, 12 (2011), 2088–2095. 2

[WTS*07] WIEBEL A., TRICOCHE X., SCHNEIDER D., JANICKE H., SCHEUERMANN G.: Generalized streak lines: Analysis and visualization of boundary induced vortices. *IEEE Transactions on Visualization and Computer Graphics, 13*, 6 (2007), 1735–1742. 2

[WYG*11] WEI J., YU H., GROUT R., CHEN J., MA K.-L.: Dual space analysis of turbulent combustion particle data. In *Pacific Visualization Symposium (PacificVis), 2011 IEEE* (2011), pp. 91–98. 2

[YP11] YANG Y., PULLIN D. I.: Geometric study of lagrangian and eulerian structures in turbulent channel flow. *Journal of Fluid Mechanics 674* (4 2011), 67–92. 1, 6

[YPI10] YANG Y., PULLIN D. I., I. B.-M.: Multi-scale geometric analysis of lagrangian structures in isotropic turbulence. *Journal of Fluid Mechanics 654* (6 2010), 233–270. 1, 2

© 2018 The Author(s)
Computer Graphics Forum © 2018 The Eurographics Association and John Wiley & Sons Ltd.

Cranfield University

CERES https://dspace.lib.cranfield.ac.uk

School of Aerospace, Transport and Manufacturing (SATM) Staff publications (SATM)

2018-08-06

On visualizing continuous turbulence scales

Liu, Xiaopei

Wiley

þÿ X i a o p e i L i u , M a n e e s h M i s h r a , M a r t i n S k o t e a n d C h i W i n g F u . O n v i s u a l i z
turbulence scales. Computer Graphics Forum, Volume 31, Issue 1, 2019, pp. 300-315
https://doi.org/10.1111/cgf.13532
Downloaded from Cranfield Library Services E-Repository