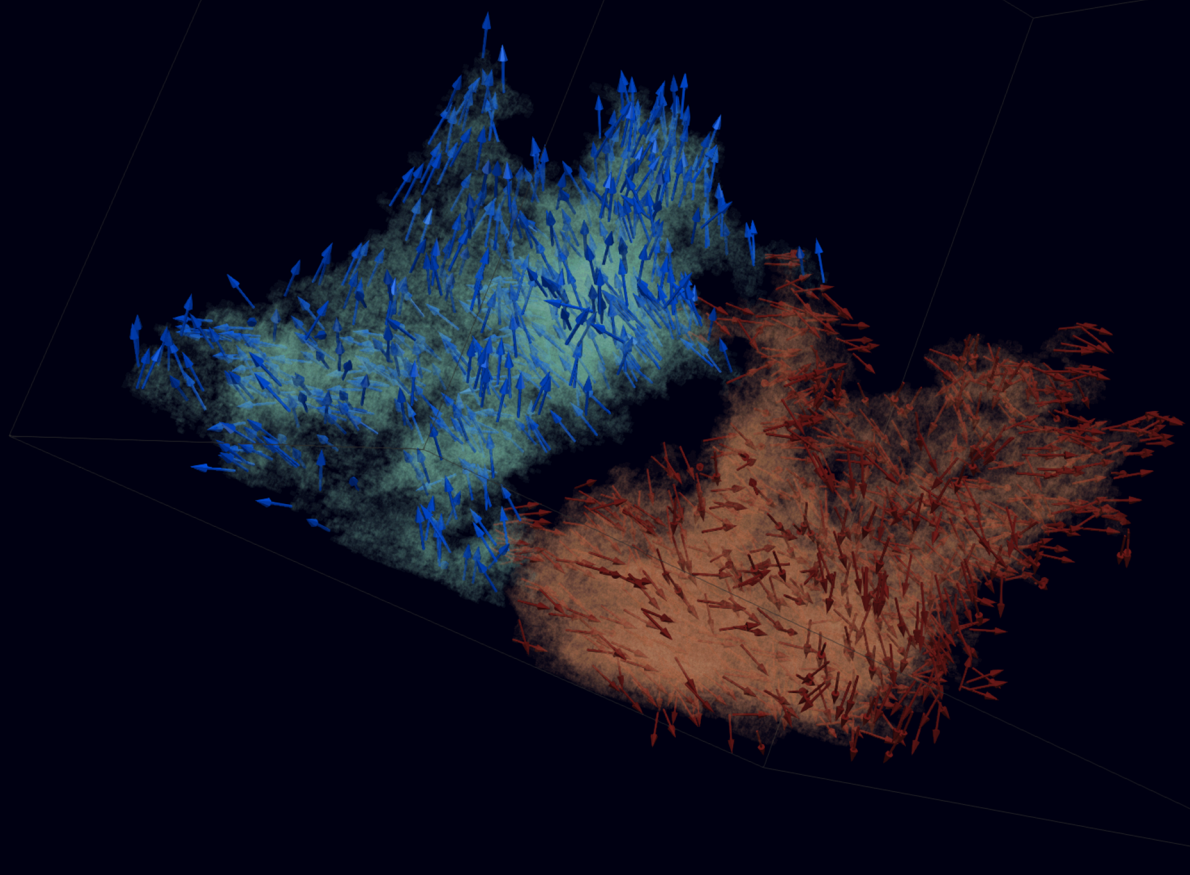
Automated Identification of Large-Scale Structures

A Clustering-Based Methodology for Homogeneous Isotropic Turbulence

Maximilian Lagorio

Delft University of Technology



TU Delft

Automated Identification of Large-Scale Structures

A Clustering-Based Methodology for Homogeneous Isotropic Turbulence

by

Maximilian Lagorio

To obtain the degree of Master of Science
at the Delft University of Technology.
To be defended publicly on 25-09-2025.

Student Number:	5850088
Thesis committee:	Dr. N.A.K. Doan (Supervisor) Dr.ir. G.E. Elsinga (Supervisor) Dr. S.J. Hulshoff (Chair) Dr.ir. M. Pini (External Examiner)
Location:	Faculty of Aerospace Engineering, Delft University of Technology

TU Delft

Acknowledgements

As this academic journey at TU Delft comes to a close. I wish to express my deepest gratitude to those who have accompanied and supported me along the way.

First and foremost, I would like to sincerely thank my thesis supervisors, Dr. N.A.K. Doan and Dr.ir. G.E. Elsinga. Thank you for proposing this thesis topic, which sparked my curiosity and passion from the very beginning. Your invaluable guidance, intellectual insights, and consistent support were essential in shaping this work. This thesis would not have been possible without your expertise and mentorship.

My heartfelt thanks to my colleagues and friends, who have been there throughout my academic journey. The countless discussions, collaborative brainstorming sessions, and shared moments of both frustration and triumph made this academic journey not just manageable, but truly enriching and enjoyable.

Finally, I wish to extend my deepest gratitude to my family. Your unwavering support, endless belief in me, and continuous encouragement have been the foundation of my success.

Maximilian Lagorio
Delft, September 2025

Abstract

Identifying large-scale coherent structures in homogeneous isotropic turbulence is crucial for advancing the understanding of turbulent phenomena, including intermittency and energy transfer. However, the current knowledge and statistical characterization of these structures remain limited due to the absence of efficient and consistent identification techniques. This thesis presents an automated methodology based on the HDBSCAN clustering algorithm to identify large-scale coherent structures.

The method successfully detects coherent large-scale structures, characterized by a quasi-uniform velocity direction. It has been tested across a wide range of Reynolds numbers ($Re_\lambda \approx 37 - 1131$), providing insights into their spatial organization and interactions. High dissipation regions were observed to occur between neighboring structures, indicating a direct link between large-scale motions and small-scale intermittency through shearing mechanisms. Additionally, the methodology was extended to time-resolved datasets, enabling temporal tracking and analysis of the evolution of these structures in physical space.

This thesis provides a robust and efficient framework for coherent large-scale structure identification in homogeneous isotropic turbulence and provides new insights into their statistical properties, dynamics, and role in turbulent energy transfer.

Contents

List of Figures

List of Tables

1

Introduction

Turbulent flows are ubiquitous in nature and engineering. From weather phenomena to the flow around airplanes. Turbulence also plays a crucial role in star formation [1], and it is hypothesized that turbulent structures are one of the main sources of noise generation [2].

Turbulence is a complex phenomenon that can be classified as *deterministic chaos*. This means that the flow is determined by known equations, initial and boundary conditions. However, small variations in these conditions can lead to large differences in the flow field due to the non-linear nature of turbulent flows. Due to this, the study of turbulent flows is a challenging task, and to this day there is no comprehensive theory that can predict the behavior of turbulent flows. Therefore, scholars often resort to heuristic approaches to predict the behavior of turbulent flows [3].

Most existing theories are based on Kolmogorov's similarity hypothesis [4], which states that the small scales of the turbulent flow are statistically independent of the large scales and therefore can be considered as statistically universal. This theory is based on the concept of the energy cascade, which describes the energy transfer mechanism in turbulence. The large scales are the most energetic, and they transfer energy to the smaller scales with a cascade process. This cascade process continues until it reaches the Kolmogorov scale η where the energy is dissipated through viscosity. The cascade process is summarized by the famous verse by Richardson [5]:

Big whirls have little whirls,
That feed on their velocity;
And little whirls have lesser whirls,
And so on to viscosity.

The Kolmogorov hypothesis assumes that energy transfer is gradual from large scales to small ones. Therefore, the small scales do not interact with the large scales directly and are assumed to be independent of the large scales. However, several studies [6, 7, 8] have shown that there are interactions between the Fourier modes that produce a non-local energy transfer between large and small scales. Moreover, recent studies showed the presence of high-dissipation regions in homogeneous isotropic turbulence [9], suggesting that the behavior of small scales is not entirely independent of the large scales. These findings promote interest in uncovering the actual physical mechanisms that govern the cascade process. Elsinga and Marusic, Elsinga, Ishihara, and Hunt [10, 9] identified high-dissipation regions referred to as intense shear layers in homogeneous isotropic turbulence. Similarly, Park and Lozano-Durán [11] identified regions of intense transfer.

These findings suggest the presence of an underlying large-scale organization of small-scale structures within the flow. The existence of large-scale structures provides a plausible mechanism for this organization, as they can influence the behavior of smaller-scale turbulent motion through mechanisms such as vortex stretching [12]. Large-scale structures are the regions of the flow that are the primary carriers of kinetic energy and momentum, with dimensions comparable to the integral length scale ($\mathcal{L}$). Identifying these structures is crucial for understanding the possible interactions between large-scale

and small-scale turbulent motions.

The identification of large-scale structures in homogeneous isotropic turbulence is a challenging task. The flow does not have any preferred direction and therefore the structures also do not have any preferential alignment. This means that the structures can be aligned in any direction, and therefore the identification of these structures is not trivial. Although initial studies have shown the presence of large-scale structures in homogeneous isotropic turbulence [11, 10], the identification of these structures is often done by hand. Consequently, it is difficult to identify those structures consistently across different datasets. Additionally, this process is currently very time-consuming and requires a lot of manual effort and hence is not suitable for a statistical characterization of the structures. Therefore, there is a need for a methodology that can automatically identify large-scale structures in homogeneous isotropic turbulence in a consistent and reproducible way.

This leads to the main research question of this thesis:

How can large-scale structures in homogeneous isotropic turbulence be identified in a consistent, reproducible, and efficient way?

To answer this question, we will develop a methodology to automatically identify large-scale structures in homogeneous isotropic turbulence. Using this approach, we aim to investigate the nature and behavior of these structures, with particular focus on the following sub-questions:

- *What are the characteristics of large-scale structures in isotropic homogeneous turbulence?*
- *How do these characteristics of these structures change with the Reynolds number?*
- *Is there any observable phenomena that can be attributed to the presence of large-scale structures?*
- *What are the observable dynamics of large-scale structures?*

This thesis is structured as follows. Chapter 2 lays the theoretical groundwork by providing essential background on homogeneous isotropic turbulence. Following this, Chapter 3 critically reviews traditional identification techniques, discussing their contributions and limitations in identifying coherent structures. Chapter 4 then explores the potential of modern machine learning methods, covering both clustering and supervised learning approaches as alternatives. The core of this work is detailed in Chapter 5, which presents the proposed methodology, from data pre-processing and HDBSCAN clustering to the post-processing steps for structure extraction. Chapter 7 presents the findings, including a detailed analysis of a high-Reynolds-number case, an investigation into the effect of the Reynolds number on the structures, and an examination of their temporal evolution using a time-resolved dataset. Finally, Chapter 8 concludes the report by summarizing the key findings in relation to the research questions and suggests directions for future work.

2

Background

Homogeneous isotropic turbulence can be seen as a simplified version of real turbulent flows. The statistical properties of the flow are invariant under translation and rotation. Homogeneous isotropic turbulence, because of its assumptions, is much less computationally expensive to simulate than real flows. This allows us to have access to simulations that are able to resolve all the relevant scales of the flow. In real flows, this is often unfeasible, since the very fine resolution required to resolve boundary layers leads to excessive computational costs and requirements. Therefore, to simulate real flows, we often resort to model fully or partially the behavior of the flow based on the insights gained from the study of homogeneous isotropic turbulence.

With the increase of available computational resources in recent years, Large Eddy Simulations (LES) became increasingly accessible, even for complex flows. LES simulations allow to resolve the large scales of the flow while modeling the small scales. This methodology allows cutting the computational cost drastically, however, it needs an accurate model of the energy transfer to the sub-grid scales in order to be accurate. Most LES sub-grid models are based on the assumption that the small scales are based on Kolmogorov's theory [13], which states that at high Reynolds numbers, the small scales are independent of the large scales. However, recent developments in the turbulence field have shown that even the small scales of the flow show local behavior. For example, Ishihara, Kaneda, and Hunt [15] identified thin shear layers composed of small-scale eddies with extremely high dissipation rates in homogeneous isotropic turbulence. This finding challenges the local isotropy assumption of Kolmogorov's theory. The presence of these shear layers indicates that the small-scale structures in turbulence are not completely independent of the large-scales. Instead, they exhibit strong spatial intermittency, suggesting that energy transfer between the scales may depend on local flow structures. In addition, Elsinga and Marusic [10] demonstrated that these sheer layers show a preferential alignment of 45° deg in the eigen-frame of the strain tensor and separate two almost uniform flow regions, which can be attributed to coherent flow structures.

One of the main requirements in correctly modeling in the energy cascade process is to understand how the energy is transferred from one scale to another. Currently, there are two main theories that describe the dynamics of the energy cascade process. The most common one is that the energy is transferred through a process known as vortex stretching, illustrated in Figure 2.1a. This theory was introduced by Taylor [16] and is based on the idea that a high vorticity region is stretched by an extensional strain field along its rotational axis reducing its size and therefore, according to the conservation of the angular momentum, increasing its vorticity. This mechanism allows the transfer of energy by converting straining motions to vortical motions of smaller size [17]. The second theory is based on the self-amplification of strain and is illustrated in Figure 2.1b. In this case, a region of strong compressive strength is intensified as the faster moving fluid regions reach the slower moving ones. This mechanism increases the compressive strain and reduces the spatial extent of the compressive region. Currently, it is not clear what is responsible for these two mechanisms, and it is still an open question in the field of turbulence. However, it is hypothesized that interactions between large-scale structures and small-scale structures might play an important role in the energy transfer process. Therefore, being able

to identify large-scale structures in homogeneous isotropic turbulence is crucial for better understanding homogeneous isotropic turbulence and the energy transfer process.

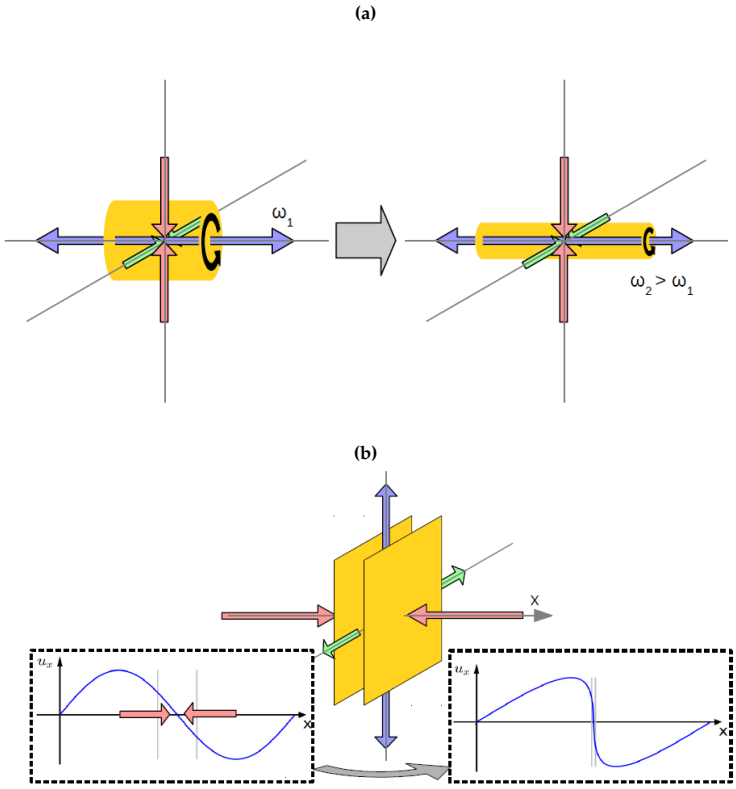


Figure 2.1: Schematic representation of the two main mechanisms of energy transfer in turbulence [17]. **(a)** Vortex stretching mechanism. **(b)** Strain self-amplification mechanism.

3

Identification techniques

Coherent structures can be defined as *'A connected turbulent fluid mass with instantaneously phase-correlated vorticity over its spatial extent'* [18]. While this definition focuses on vorticity, it can be easily extended to other flow variables such as velocity and flow direction. Therefore, in this thesis, coherent structures will be considered as regions of the flow where a variable is not random, but exhibits a certain degree of uniformity. This definition has been applied to different variables and has been used to obtain different types of coherent structure. For example, Elsinga and Marusic [10] applied it to the velocity direction and Park and Lozano-Durán [11] applied it to the energy transfer.

Furthermore, Adrian [19] highlighted that spatial coherency is not a sufficient condition to define a coherent structure; it should also show a certain degree of temporal coherency. In other words, for a flow region to be considered a coherent structure, it should also persist for long times.

A lot of research has been done in the field of coherent structures in wall bounded flows. However, as of today, there is very little research on large-scale coherent structures in homogeneous isotropic turbulence due to the challenges involved with their identification.

This chapter covers the current methods available in literature to identify coherent structures in turbulent flows.

3.1. Coherent structures in wall bounded flows

Coherent structures in wall-bounded flows have been studied for a long time. An initial identification of coherent structures can be traced back to the work of Kline et al. [20] in 1967. Using hydrogen bubble visualization in a water channel, they identified the existence of low-speed streaks originating in the near-wall region. These streaks were observed to undergo a gradual process of lift-up and ejection.

Another key structure found in the logarithmic layer of wall-bounded turbulent flows is the hairpin vortex. These vortices commonly appear in packets that significantly influence the turbulence statistics [19]. These packets exhibit a quasi-uniform momentum aligned with the stream-wise velocity and are on the order of the boundary layer thickness δ. Although they occupy only a small fraction of the flow volume, these structures have been shown to have a substantial impact on the flow's energy transfer [19].

While coherent structures in wall-bounded flow have been extensively studied, research regarding coherent structures in homogeneous isotropic turbulence remains sparse. This can be attributed to the fact that the structures in isotropic turbulence lack several properties that are present in wall-bounded flows. Mainly, in wall-bounded flows the presence of a mean flow makes the structures more elongated and aligned with the mean flow, which makes their identification easier. In isotropic turbulence, the lack of this preferential direction makes the structures have a less defined shape.

3.2. Vortex identification techniques

Vortex identification techniques are a set of methods that are used to distinguish regions where rotations dominate over strain. Various criteria have been proposed over the years to identify these regions. The Q criterion, the lambda-2 criterion, and the swirling strength [21] are some of the most common vortex identification techniques. All of these techniques compute a variable for all points in the flow field based on the velocity gradient tensor. After that, the vortices can be extracted by an empirical threshold or by a mathematical cut-off of the variable. Due to the threshold being empirically chosen, often those techniques mistakenly identify regions of high strain as vortices. More modern methods are available that are more robust to the threshold choice called *third generation vortex identification techniques* like the Omega-Liutex [22].

All these methods are based on the velocity gradient tensor, which works well for small-scale structures, since they are associated with very high gradients. However, large-scale structures are highly uniform, and therefore the velocity gradient tensor is not informative. For this reason, these methods are not suitable for the identification of large-scale structures.

3.3. Modal analysis

Modal analysis is a technique that is used to identify the dominant modes in a flow field. Proper Orthogonal Decomposition (POD) [23] is a modal analysis technique that decomposes the flow into a set of orthogonal modes ordered based on the energy content. This technique has become very popular for studying the dynamics of turbulent flows, and it allows to identify the most energetic structures in the flow. While POD works well to identify the most energetic modes, it does not directly provide information regarding the structures. For example, a single structure could be represented by multiple modes, or a mode could be composed of different structures at different spatial scales. This makes extracting structures from POD modes a challenging task. New approaches like mPOD [24] incorporate multiresolution analysis to extract modes in specific scale ranges.

Another modal analysis technique is the Dynamic Mode Decomposition (DMD). This method computes the modes of the flow based on its temporal evolution. Each mode is associated with a frequency and a growth rate. However, in order to compute the dynamic behavior, this method would require to load in memory all the snapshots of the flow. This is not feasible for large-scale simulations and to overcome this issue a new method called streaming DMD was proposed [25]. While DMD is primarily used for dynamic analysis, it has also been used to identify long-lived structures in turbulent flows. Furthermore, it has been shown that the structures obtained from DMD are equivalent to those extracted using POD [26].

Both POD and DMD are able to identify persistent energetic modes in flows. While these methods manage to extract the most energetic modes, there is no clear link between those and the coherent structures. Furthermore, POD modes have been shown to degenerate into Fourier modes for homogeneous isotropic turbulence [27]. In this context, POD does not provide significant benefits over Fourier analysis while requiring greater computational effort.

3.4. Thresholding techniques

An initial identification of large-scale structures is attributed to the work of Siggia [28], who applied an arbitrary vorticity threshold to extract coherent features such as vortex tubes, sheets, and blobs from DNS simulations (Figure 3.1a). Although this work was groundbreaking at the time, the Reynolds number based on the Taylor microscale (Re_λ) was only 100, which is too low for a fully developed energy cascade. Consequently, the structures identified in their work may not be representative of the structures present in high Reynolds number flows.

A similar thresholding method was later employed by Moisy and Jiménez [29], who applied combined vorticity and strain-rate thresholds at a marginally higher Reynolds number ($Re_\lambda \approx 168$). The study managed to identify similar structures to those identified by Siggia [28] (Figure 3.1b). In addition, Moisy and Jiménez applied box-counting to characterize the structures and found that regions of intense strain and vorticity are not randomly distributed, but instead form clusters with sizes on the order of the inertial range, implying that small-scale intermittency might be modulated by large-scale organization.

More recently, Ishihara, Kaneda, and Hunt [14] introduced a coarse-graining procedure to suppress small-scale noise in the flow field at $Re_\lambda \simeq 1131$. After coarse-graining the enstrophy field, a threshold was applied to identify the large-scale structures. Using this method, the authors were able to identify sheet like structures (Figure 3.1c) with widths of the order of the integral scale and thickness of the order of the Taylor microscale. Furthermore, these structures were shown to have dissipation rates significantly higher than the domain average, suggesting a potential role in the energy cascade.

Despite their contributions, these methods depend heavily on threshold selection, which is often determined a posteriori through flow visualization. As a result, the structures are sensitive to the chosen threshold value, which hinders reproducibility and automation. Furthermore, the structures identified by these approaches often include at least one dimension on the order of the Taylor or Kolmogorov scale, which makes it difficult to classify them as large-scale structures. Another limitation of this method is that, relying on a vorticity threshold, it does not enforce any uniformity in the identified structures and thus lacks the properties necessary to be considered coherent.

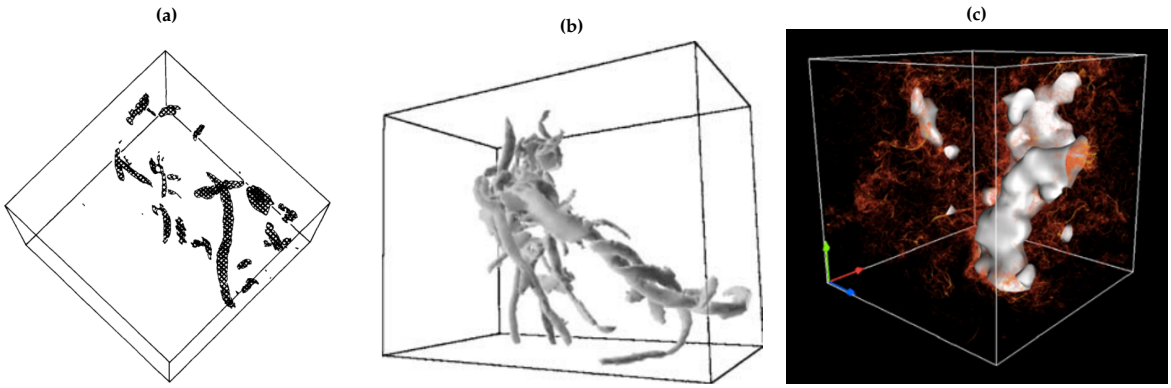


Figure 3.1: Examples of structures identified using a threshold. **(a)** structures identified by Siggia [28]. **(b)** structures identified by Moisy and Jiménez [29]. **(c)** structures identified by Ishihara, Kaneda, and Hunt [14].

3.5. Transform-based methods

Transform-based methods such as Fourier transforms have been extensively used to study and characterize the multiscale nature of turbulent flows [30]. These methods are based on the idea that the flow field can be decomposed into a set of basis functions. In the case of the Fourier transform, the decomposition is performed using a predefined set of sinusoidal basis functions, known as Fourier modes, each corresponding to a specific frequency.

The idea behind this method is that small-scale structures, which show rapid changes in flow variables over short distances, correspond to high-frequency modes. Similarly, large-scale structures, which exhibit slower spatial variation, are captured by low-frequency modes. This approach is based on the analysis of the turbulent energy spectrum, illustrated in Figure 3.2, which shows that most of the energy is concentrated in the low-frequency range, while the energy gradually decreases at higher frequencies [30].

Based on this idea, various researchers have proposed different methods to identify structures by reconstructing the field using only a few modes. For example, Goto [31] identified vortex tubes (Figure 3.3a) by applying a low-pass filter with a sharp cut-off to the vorticity field in spectral space for $Re_\lambda \approx 187$. Similarly, Leung, Swaminathan, and Davidson [32] and Doan et al. [33] applied a band-pass filter to detect enstrophy and dissipation structures (Figures 3.3b and 3.3c) for a range of Re_λ.

While these methods are capable of identifying turbulent structures, the structures are reconstructed from a filtered field composed of smooth basis functions. This implies the assumption that a structure occupies only a limited range of scales in Fourier space. However, this is not always the case. For instance, a shear layer may contain sharp features in some regions, which are typically associated with high-frequency modes. As a result, the structure would be represented by a broad range of frequencies in the spectral domain. In addition, structures at different scales require different filter cut-off values

which are usually flow dependent.

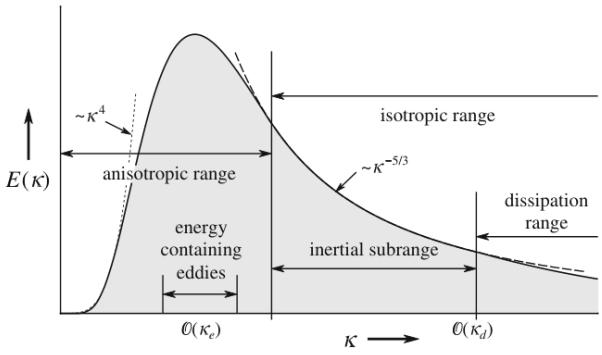


Figure 3.2: Example of turbulent energy spectrum [3]. $E(\kappa)$ is the energy content of the flow at wave number κ.

To address these limitations, methods were developed based on wavelet transforms [27, 34] and curvelet transforms [35], which use localized basis functions instead of global ones. Both of these methods are multiscale and, therefore, allow for better structure identification. In addition, curvelet transforms are also directional, allowing for an easier identification of elongated structures. While these methods are capable of considering multiple scales simultaneously, structures are still extracted by applying a threshold to the transformed field. This leads to similar issues as in Fourier methods, making it difficult to identify specific features of the structures and, therefore, introducing assumptions regarding the structure shapes.

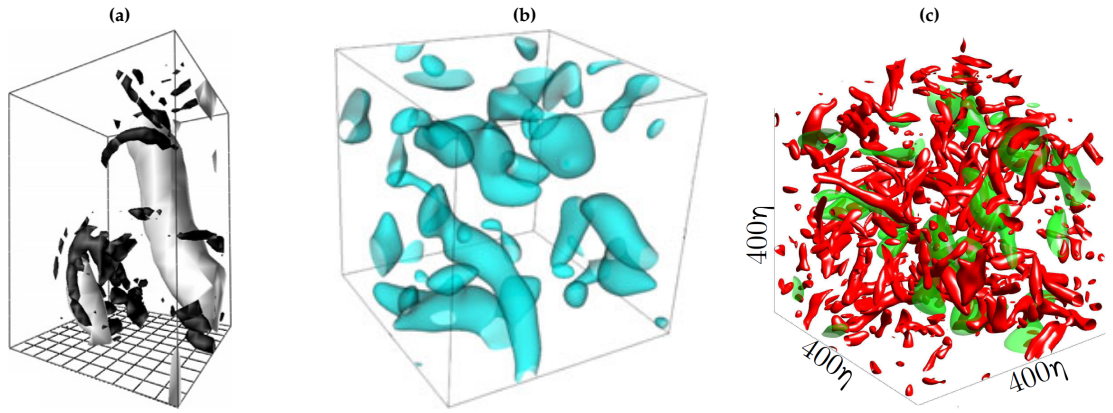


Figure 3.3: Examples of structures identified using transform-based techniques. **(a)** structures identified by Goto [31]. **(b)** Structures identified by Leung, Swaminathan, and Davidson [32]. **(c)** Structures identified by Doan et al. [33].

3.6. Histogram method

The most recent and applicable method for identifying large-scale structures in isotropic turbulence is the procedure illustrated by Elsinga and Marusic [10]. This method is based on the idea that large-scale structures are characterized by a quasi-uniform velocity, meaning they correspond to regions of the flow where the velocity vectors are aligned in a similar direction. Leveraging this concept Elsinga and Marusic converted all points to spherical coordinates and then computed a 2D histogram of the azimuth and elevation angles. Since large-scale structures are characterized by quasi-uniform velocity and occupy large volumes, the histogram will exhibit a peak in the regions where these structures are present. Regions marking the histogram peaks were manually identified based on visual inspection, as illustrated in Figure 3.4, and extracted.

While this method is effective in identifying large-scale structures, it relies on the manual selection

of histogram peaks that introduce subjectivity. Additionally, the method uses spherical coordinates, which deform bin sizes when projected onto Cartesian space. This results in a non-uniform distribution of bins, making peak identification more challenging. To compensate for this distortion, the authors applied a scaling factor equal to the cosine of the elevation angle to adjust the bin values. However, even with this correction, the bins may not accurately represent the underlying velocity direction distribution. Furthermore, the method is sensitive to the choice of bin size: Smaller bins may lead to an increased number of peaks, while larger bins may smooth out relevant features and may result in the merging of peaks.

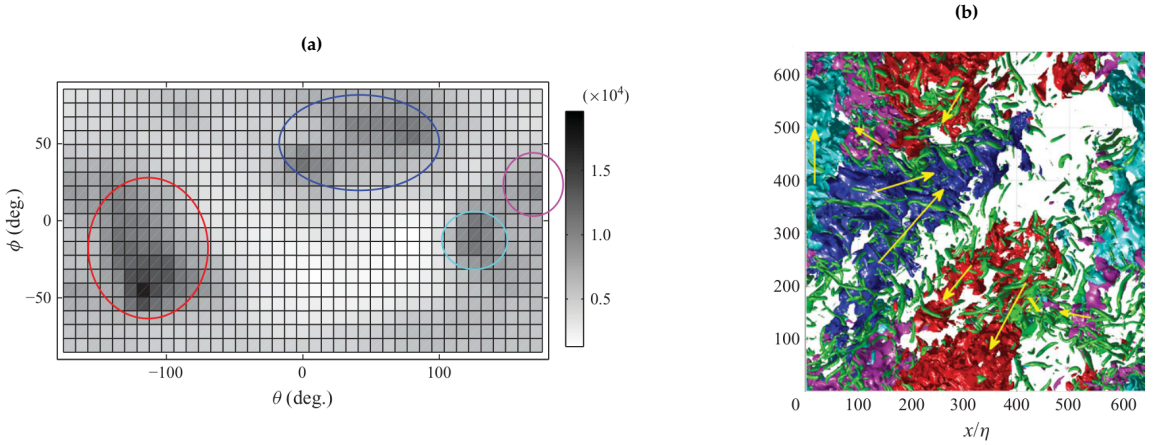


Figure 3.4: Structure identification via histogram method [10]. **(a)** Identification of histogram peaks. **(b)** Extracted structures.

A first attempt to automate the peak selection process was made by A.K.M. Ramanna [36] through instance segmentation. To overcome the limitations posed by periodic boundary conditions and non-uniform binning due to elevation stretching, the method incorporates coordinate translation and symmetry operations. This results in four different histograms, which are then analyzed using a U-Net [37] to identify the peaks. While this method is capable of automating the peak selection process, it still relies on the histogram method to identify the structures. This means that the method is still sensitive to the bin size. Additionally, the U-Net requires training data, which is not sufficiently available for the current case. The author trained the U-Net using synthetic data generated from Gaussian peaks with noise. However, the training masks are dependent on a hyperparameter that defines the selection radius around the peak. This parameter and the artificial training data make assumptions regarding the peak shape and size which could make the detection of irregularly shaped peaks more challenging and less accurate.

3.7. Concluding remarks

In this chapter the available methods to identify coherent structures in turbulent flows were reviewed, with an emphasis on the added difficulty of finding large-scale organization in homogeneous isotropic turbulence (HIT). We adopted a practical definition of a coherent structure as a connected flow region that is not random but shows some spatial uniformity and, when possible, temporal persistence [18, 19].

Classical vortex criteria (Q, λ_2, swirling strength, Ω–Liutex) isolate compact vortexes but lose effectiveness when gradients are small, as is typical of large-scale motions in HIT. Modal approaches (POD, mPOD, DMD, streaming DMD) reveal high-energy structures. However, there is no clear link between those structures and coherent regions in HIT flows.

Thresholding on vorticity, strain, enstrophy, or dissipation has provided useful insights across different Reynolds numbers, but remains highly sensitive to user-defined cutoffs that must be chosen a posteriori. Transform-based techniques, such as Fourier, wavelet, or curvelet transforms, can target specific scales and orientations but still rely on user-defined thresholds and assume that structures occupy a limited range of scales, therefore introducing implicit assumptions about their shape.

The histogram method offers a promising approach for identifying large-scale structures in homogeneous

isotropic turbulence. However, it introduces subjectivity through manual peak selection and remains sensitive to the chosen bin size. Efforts to automate this process by A.K.M. Ramanna [36] using U-Nets show potential, but still face challenges related to bin size sensitivity, spherical coordinate projection, and shape assumptions arising from the use of artificially generated training data.

Overall, an effective method to identify large-scale structures in HIT flows should:

1. Avoid making assumptions about the shape of the structures.
2. Use parameters that are independent of the specific flow conditions and have a well-defined effect on the identified structures.
3. Not be reliant on user-defined thresholds or parameters that require a posteriori tuning.

4

Machine learning methods

Recent advancement in machine learning have opened new possibilities for the identification of coherent structures in turbulent flows. Traditional approaches, as discussed previously, often rely on manual thresholding and heuristic criteria, which limit reproducibility and automation. Using machine learning techniques, it is possible to automate the identification process which would allow a statistical characterization of large-scale structures. In addition, with machine learning it is possible to identify structures that are not easily identifiable using traditional methods. For example, in the case of the histogram method treated in Section 3.6, it is possible that some structures have a very faint peak on the histogram, which would make it difficult to identify visually.

This chapter explores the current machine learning techniques that could be applied to the identification of large-scale structures in homogeneous isotropic turbulence. The chapter is divided into two main parts: the first part focuses on clustering techniques, while the second part covers supervised learning techniques. Clustering techniques are primarily used to identify regions of the flow that exhibit similar properties, while supervised learning techniques are employed to classify these regions based on training data. The chapter concludes with a discussion of the potential applications of these techniques in the identification of large-scale structures.

4.1. Clustering techniques

Clustering is a data analysis technique that groups data points together in a way that the points in the same group are more similar to each other than to those in other groups by some sort of metric. Those groups are called clusters, however, the term cluster is not well-defined in literature [38]. Different clustering techniques use different definitions, and thus the clusters obtained by those techniques vary significantly in their properties. For example, some clustering techniques are based on the idea that a cluster is a set of points that are close to a center point, while others define a cluster as a set of points that are connected to each other in dense regions in the feature space.

The main idea behind using clustering techniques to identify large-scale structures is based on the assumption that these structures are characterized by a certain degree of uniformity. This means that the flow variables in the region of the structure should be similar to each other. Therefore, clustering techniques can be used to identify regions of the flow that exhibit similar properties. Once these regions are identified, they can be further analyzed to extract the structures.

4.1.1. K-means

K-means is a clustering algorithm that partitions the data into k clusters. The k-means algorithm is centroid based, meaning that it defines a cluster by defining its center and assigns points to that cluster in a way that minimizes the deviation from the center of the cluster [39]. The algorithm works by first selecting randomly k initial cluster centers from the dataset. After that, the k-means algorithm iteratively assigns the points to the cluster and then updates the centroid to the mean value of the cluster until the centroids do not change significantly.

The main advantage of the k-means algorithm is its simplicity and speed. However, this technique has several limitations. For example, the number of clusters k needs to be specified beforehand. While there are numerous ways to determine an appropriate number of clusters like silhouette scores [40] it would still require some user input in the detection process. Additionally, k-means does not have a way to identify noise or outliers, as it partitions all data points into hyper-spherical clusters [39].

4.1.2. Mutual K-nearest neighbors

Mutual K-nearest neighbors (M-KNN) [41] is a graph-based clustering technique. The M-KNN algorithm works by first identifying the k nearest neighbors of each point in the dataset by computing the pairwise distance between all points. The points that have smaller distance to each other are then merged into an initial cluster. The clusters are then expanded and merged based on the distance between the points and on how many points are connected between the clusters. The process continues iteratively until a certain criterion is met, such as the desired number of clusters (λ).

While M-KNN requires two different parameters, the number of neighbors (k) and the desired number of clusters (λ), it is important to note that the number of clusters is not a hard threshold. Therefore, depending on the connectivity of the point, it is possible to obtain more clusters than the desired number. This makes M-KNN a flexible clustering technique that can adapt to the data. Additionally, it has been shown that M-KNN is less sensitive to the parameter choice than other clustering techniques like DBSCAN or k-mean [41, 42].

M-KNN has been successfully applied to the identification of coherent structure in turbulent flow [42]. The authors applied the M-KNN algorithm to particles trajectories and spatial proximity of the particles. This approach uses Lagrangian tracking of the particle to identify the coherent structures. While this approach is able to identify coherent structures, it requires time-resolved simulations. However, datasets of homogeneous isotropic turbulence are typically generated through direct numerical simulations and are often extremely large. This makes loading multiple time steps in memory unfeasible.

4.1.3. DBSCAN

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [43] is a clustering technique that defines clusters as dense regions in the feature space. The main idea behind this technique is that a cluster is a set of points that are close to each other and separated from other clusters by regions of lower density. This means that the clusters do not have a predefined shape, which makes DBSCAN suitable for identifying irregularly shaped peaks. Furthermore, DBSCAN does not require the number of clusters to be specified a priori, which makes it appropriate for identifying structures in flows where the number of structures is unknown.

The DBSCAN algorithm requires two parameters: the neighborhood radius (ϵ), which defines the distance within which points are considered neighbors, and the minimum number of points ($minPts$) required to form a cluster. The algorithm works by labeling all points in the feature space as core points, border points, or noise points. Core points are those that have at least $minPts$ neighbors within the ϵ radius, while border points are those that are within the ϵ radius of a core point but do not have enough neighbors to be considered core points. Noise points are those that are not within the ϵ radius of any core point. Once the points are labeled, the algorithm groups all core points and their neighbors into clusters. An example of the DBSCAN algorithm is shown in Figure 4.1. In this example, A is a core point, B and C are border points, and N is a noise point. The algorithm groups A, B, and C into a cluster and labels N as noise.

The main advantage of DBSCAN is that it can identify clusters by filtering out noise points, which makes it suitable for turbulent flows where a high level of background fluctuations is expected. While DBSCAN is a powerful clustering technique, it is very sensitive to the choice of the neighborhood radius (ϵ) [45, 46, 47]. The choice of ϵ can significantly change the clusters obtained. For example, using a small ϵ could fragment a large but less dense cluster into multiple clusters, or if they do not satisfy the $minPts$ condition, they could be labeled as noise. On the other hand, using a large ϵ could merge multiple clusters into a single cluster [47]. This parameter is hard to determine beforehand due to its sensitivity to the dataset and its limited interpretability. Additionally, DBSCAN uses a fixed global ϵ and therefore is not able to identify clusters in datasets with very different densities or nested clusters [45].

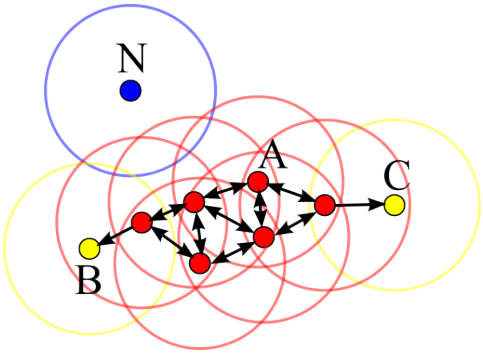


Figure 4.1: Example of the DBSCAN algorithm. The points A, B and C are grouped into a cluster while the point N is labeled as noise [44].

4.1.4. HDBSCAN

Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) [45] is a hierarchical extension of DBSCAN that addresses some of its limitations. While DBSCAN uses a fixed ϵ, HDBSCAN spans all possible ϵ values and builds a hierarchy of clusters. This allows HDBSCAN to identify clusters at different scales and to merge or split clusters based on their stability. The HDBSCAN algorithm works by first computing the core distance (d_{core}) for each point, which is the distance to the $min\_samples$ nearest closest points. The algorithm then builds a Mutual Reachability Graph, where the nodes are the points in the dataset and the edges are the mutual reachability distances between the points defined as:

$$d_{mreach}(\mathbf{x}_a, \mathbf{x}_b) = \max\{d_{core}(\mathbf{x}_a), d_{core}(\mathbf{x}_b), d(\mathbf{x}_a, \mathbf{x}_b)\} \tag{4.1}$$

Here, $d(\mathbf{x}_a, \mathbf{x}_b)$ is the distance between the points $\mathbf{x}_a$ and $\mathbf{x}_b$. Following this, a minimum spanning tree (MST) is built from the Mutual Reachability Graph. The core distances of each point are also included as self-edges, which represent the density level at which each point can be considered its own cluster. To build the cluster hierarchy, the algorithm sorts all of these edges (from both the MST and the self-edges) by their weights in increasing order. Then it uses this sorted list to create a hierarchy of merging clusters.

Once the hierarchy is built, the lower levels of the hierarchy are composed of very small and short-lived clusters. To overcome this issue, a minimum cluster size ($min\_cluster\_size$) is defined. This parameter defines the minimum number of points required to form a cluster. The algorithm then checks each level of the hierarchy, and when a split occurs, it checks if the resulting clusters contain at least $min\_cluster\_size$ points. If a cluster does not satisfy this condition, the points within that cluster are labeled as noise, condensing the large and complex hierarchy into a more manageable and smaller tree, as illustrated in Figure 4.2. The hierarchy is visualized using a dendrogram where the y-axis represents the lambda (λ) value, which is the inverse of the local density, which means that high values λ correspond to large distances and low-density regions, while low λ values represent small distances and high-density clusters.

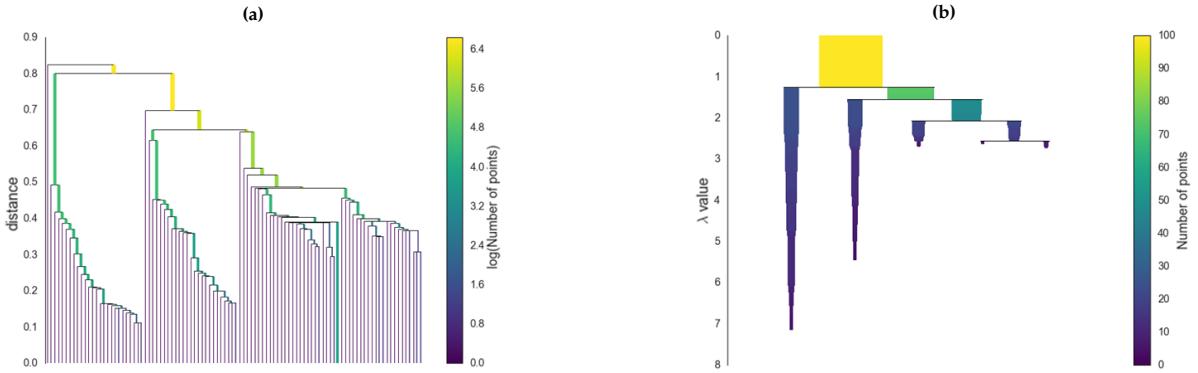


Figure 4.2: An example of the hierarchy condensation process [48]. The full hierarchy (**a**) is simplified into the condensed hierarchy (**b**) by removing noise points and merging nearby clusters.

Once the condensed hierarchy is obtained, there are two possible approaches to extract the clusters. The first approach is to select the final groups at the end of each branch in the hierarchy, known as the leafs, which are the stable clusters that do not split any further. This approach yields clusters that are more homogeneous and compact while still allowing variable density within the clusters [48]. The second approach is to select a set of non-overlapping clusters by maximizing the sum of the stability scores of the clusters. The stability score of a cluster can be seen as a measure of how much the cluster persists throughout the hierarchy (decreasing ϵ). This concept was formalized in the original paper by Campello, Moulavi, and Sander [45] and is defined by adapting the notion of *excess of mass* [49]. In the case of HDBSCAN, the stability score of each cluster is defined as follows:

$$S(C_i) = \sum_{x_j \in C_i} \left(\frac{1}{\varepsilon_{\min}(x_j, C_i)} - \frac{1}{\varepsilon_{\max}(C_i)} \right) \tag{4.2}$$

Here, C_i denotes a cluster, $\varepsilon_{\max}(x_j, C_i)$ represents the smallest mutual reachability distance at which the point x_j belongs to the cluster C_i, and $\varepsilon_{\max}(C_i)$ is the maximum reachability distance beyond which the cluster either splits or is considered noise. Using these stability scores, the algorithm selects the most prominent and non-overlapping set of clusters by maximizing the sum of the stability scores [45].

HDBSCAN relies mainly on two user-defined parameters: the minimum cluster size ($min\_cluster\_size$) and the minimum samples ($min\_samples$). The $min\_cluster\_size$ parameter defines the minimum number of points required to form a cluster, while the $min\_samples$ parameter defines the number of neighbors required to compute the core distance. A good choice for the $min\_samples$ parameter is to set it equal to the $min\_cluster\_size$ parameter. This choice ensures that the core distance is computed based on the same number of points required to form a cluster, which helps to avoid noise points being classified as core points [45]. The $min\_cluster\_size$ depends on the dataset and the expected size of the clusters. However, the effect of this parameter is known beforehand and can be easily set based on a priori knowledge of the dataset. For example, in the case of large-scale structures identification in turbulent flows, the expected size of the clusters is generally hypothesized to be of at least the order of the integral scale.

4.1.5. Gaussian Mixture Model

Gaussian Mixture Models (GMM) are a probabilistic clustering technique that assumes the data is generated from a mixture of several Gaussian distributions. Each Gaussian distribution is characterized by its mean and covariance matrix. The GMM algorithm works by iteratively estimating the parameters of the Gaussian distributions and assigning each data point to the distribution that best fits it. The algorithm uses the Expectation-Maximization (EM) algorithm to estimate the parameters of the Gaussian distributions [50]. The EM algorithm consists of two steps: the expectation step and the maximization step. In the expectation, the algorithm computes the probability of each data point belonging to each Gaussian distribution based on the current estimates of the parameters. In the maximization, the algorithm updates the parameters of the Gaussian distributions based on the probabilities computed in the expectation.

The Gaussian Mixture Model is very similar to the k-means algorithm. However, instead of assigning a point to each cluster, the GMM assigns a probability of belonging to each cluster. This allows some flexibility when extracting a cluster. For example, in the case discussed in Section 3.6, running the GMM model on the azimuthal and elevation angle would make each mean correspond to the center of the peak and it would be possible to extract all the points within a standard deviation of the mean. While this can be seen as a way to handle noise points, it is important to note that the GMM is very sensitive to outliers. This is because the GMM assumes that the data is generated from a mixture of Gaussian distributions, and extreme values can significantly affect the estimates of the parameters [39]. Therefore, it is important to preprocess the data and remove any outliers before applying the GMM algorithm. Additionally, the GMM requires the number of clusters to be specified beforehand.

4.2. Convolutional neural networks

Convolutional neural networks (CNN) are a type of supervised learning that is used primarily for image classification and segmentation. CNNs require a training dataset with labeled data to learn to perform

identification tasks. A possible application is to use CNNs to identify the peak region in the histograms Section 3.6. The CNN would be trained using a dataset of histograms with labeled peaks. However, as of now, there is no available dataset.

More modern CNNs architecture can provide higher accuracies with less training data [51]. However, one of the main challenges of using CNNs is that the velocity direction, in this case defined by the azimuthal and elevation angles, is defined in a spherical coordinate system. This means that the projection deformation to Cartesian coordinates and the periodicity of the azimuthal angle must be taken into account.

4.2.1. U-Net and subsequent developments

U-Net is a type of convolutional neural network (CNN) that was originally developed for medical image segmentation [37]. The U-Net architecture consists of an encoder-decoder structure, where the encoder captures the context of the input image and the decoder enables precise localization, as illustrated in Figure 4.3. The U-Net architecture has been widely adopted in various fields thanks to its multiscale feature extraction capabilities. This architecture has already been applied to the identification of large-scale structures in turbulent flows [36]. The author used a U-Net to identify the peaks in the histogram of the azimuthal and elevation angles. The U-Net was trained using synthetic data generated from Gaussian peaks with noise. However, the training process requires a hyperparameter that defines the selection radius around the peak. This parameter makes assumptions regarding the peak shape and size, which could make the detection of irregularly shaped peaks difficult.

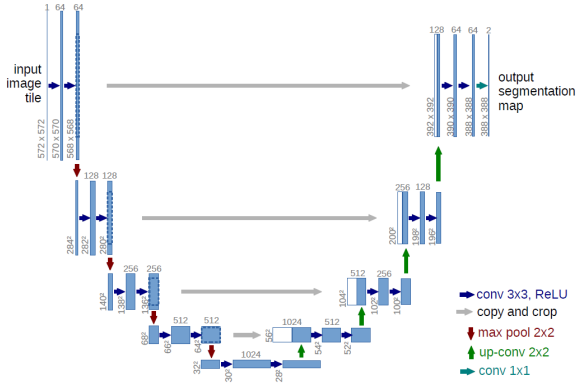


Figure 4.3: U-Net architecture [37].

As result of the U-Net's popularity, several variations of the architecture have been proposed to improve its performance. For example, U-Net++ [52] introduces nested skip pathways, which show improved segmentation performance in various applications. Another variation is the DUCK-Net [53], which incorporates residual blocks and modifies the skip connections and down-sampling operations. The DUCK-Net architecture has demonstrated impressive accuracy in polyp segmentation tasks, achieving state-of-the-art results on multiple benchmark datasets, even when trained on a very limited number of images. Additionally, the U-Net architecture has been adapted for 3D volumetric data [54]. The U-Net architecture is a powerful tool for image and volume segmentation tasks, and its flexibility and adaptability make it suitable for a wide range of applications, including the identification of large-scale structures in homogeneous isotropic turbulence.

4.2.2. Mask-RCNN

Mask-RCNN is a type of convolutional neural network (CNN) that extends the Faster R-CNN architecture [55] by adding a branch for predicting segmentation masks. The architecture consists of three main components: a backbone network, a Region Proposal Network (RPN), and a head. The backbone network is responsible for feature extraction, while the RPN generates candidate object proposals. The head then generates segmentation masks for each region of interest (ROI). Mask R-CNN is available with pretrained weights on various datasets, such as COCO [56]. Leveraging these pretrained models can significantly reduce both training time and the amount of data required to achieve good

detection accuracy.

The Mask-RCNN architecture can provide great accuracy in instance segmentation task and very high performance. Therefore, it is a good candidate for the identification of large-scale structures in turbulent flows via peak detection in the velocity angles histograms.

4.2.3. YOLO

You Only Look Once (YOLO) [57] is a real-time object detection architecture which in its original version is based on a single convolutional neural network to simultaneously predict bounding boxes and class probabilities from images. Thanks to its simplicity and speed, YOLO has become one of the most popular object detection architectures. The YOLO architecture has seen various improvements and iterations in recent years, with each improving its speed and reliability. Similarly to the Mask-RCNN, YOLO is available with pre-trained weights on various datasets making it easy to fine-tune for specific applications.

While YOLO originally focused on bounding box detection, more modern versions of the architecture like YOLOv7 [58] have been adapted for polygon detection, making it suitable for instance segmentation tasks. The YOLOv7 architecture has already been applied to the identification of large-scale structures by A.K.M. Ramanna [36]. However, the model failed to correctly identify the peaks in the velocity angle histograms and erroneously labeled regions without any discernible peak.

4.2.4. Spherical Geometry Considerations

Most CNN methods are designed to work on 2D or 3D Cartesian coordinates. However, in the case of the velocity angles, the data is defined on a spherical manifold where there is no consistent and regular neighborhood definition and therefore no straightforward convolution and pooling operations [59]. Consequently, most authors resort to project the data onto a Cartesian grid, which introduces distortion in the data. While this approach is effective, it can lead to loss of information, usually projection distorts either the angles or the area of the spherical surface. For example, Elsinga and Marusic [10] used a cylindrical projection which distorts the area of the bin. In order to correct for this distortion, the authors applied a scaling factor equal to the cosine of the elevation angle to adjust the bin values. However, even with this correction, histogram peaks associated with large-scale structures may still be distorted, particularly near the poles. Additionally, the projection introduces periodicity in the azimuthal angle, which can cause the peaks to be divided at the boundaries.

To overcome these limitations, A.K.M. Ramanna [36] used four different histograms obtained by applying coordinate translation and symmetry operations. By running the detection on all four histograms and merging the results, the author was able to mitigate the effect of the distortion. Similarly, an alternative approach is to use cube maps [60] which projects the sphere onto the inside of a cube, as illustrated in Figure 4.4, to reduce the distortions even more.

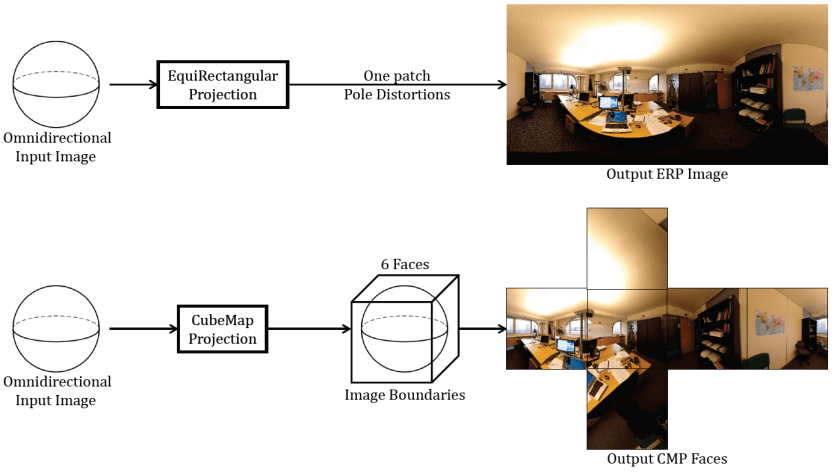


Figure 4.4: Comparison between equirectangular projection and cube map projection [61].

Another approach could be to compute the histogram on a Hierarchical Equal Area isoLatitude Pixelisation (HEALPix) [62] grid. The HEALPix grid subdivides the sphere into a set of equal area distorted rhombic dodecahedrons. This approach would allow for a projection into 2D space while preserving the area of the bins and therefore not requiring any correction. While the area of the HEALPix dodecahedrons remains constant across the sphere, their shapes become increasingly distorted near the poles. However, distortion affects angular relationships and not the weights of the bins. Therefore, the HEALPix grid would not require any correction for the bin values.

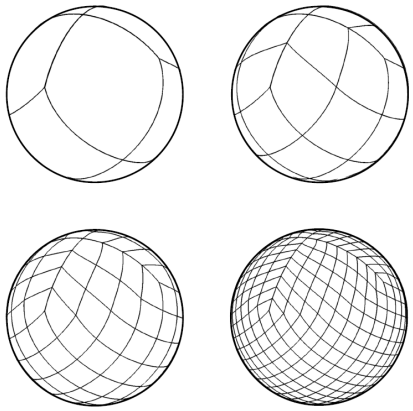


Figure 4.5: HEALPix sphere tessellation [63].

Beyond using different types of projection, a more recent application is to define the convolution and pooling operations directly on the sphere. Zhao et al. [59] proposed a U-Net architecture that works on a spherical triangular mesh. The architecture is illustrated in Figure 4.6 and was used to perform instance segmentation on cortical surfaces obtained through MRI. Using this approach, it would be possible to compute the histograms directly on the spherical surface without the need for any projection. This would allow for a more accurate representation of the data and would not require any correction for the bin values.

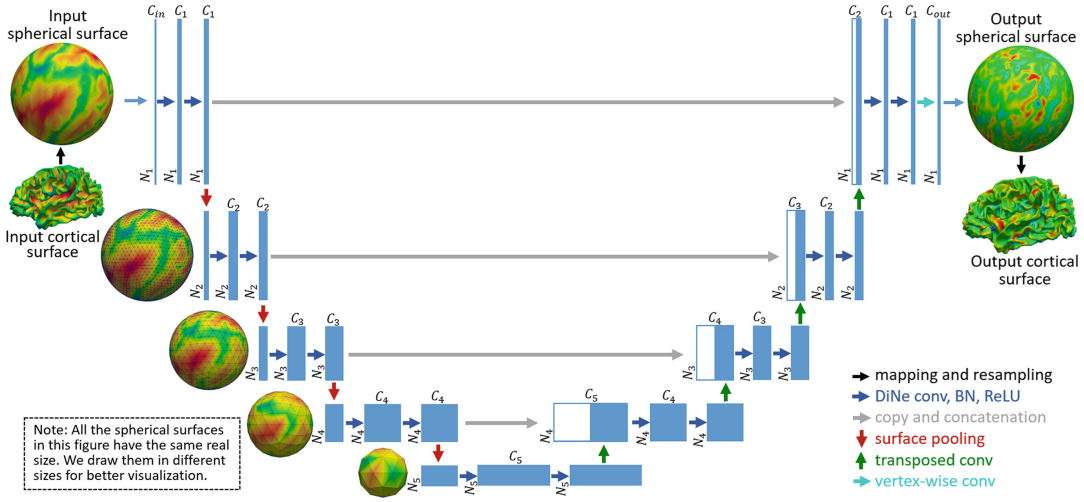


Figure 4.6: Spherical U-Net architecture [64].

4.3. Vision transformers

Vision Transformers (ViT) [65] are a type of deep-learning architecture that has gained popularity in recent years. ViTs are based on the transformer architecture developed initially for language processing tasks [66]. The main difference between ViTs and CNNs is that ViTs do not rely on convolutional layers

to extract features from the input data. Instead, ViTs use self-attention mechanisms [65] to capture long-range dependencies and relationships between different parts of the images, allowing ViTs to learn more complex representations of the data.

While ViTs consistently outperform CNNs in large datasets [67], they usually require a larger amount of training data to achieve good performance. While this can be a limitation for turbulent flows due to the lack of available datasets, the recent development of pre-trained ViTs on large datasets [51] has made it possible to fine-tune ViTs for specific applications with small datasets. Therefore, using transfer learning ViTs can become a good alternative to CNNs for the identification of large-scale structures in turbulent flows.

4.3.1. Segment Anything Model

Segment Anything Model (SAM) [68, 69] is a recent development in the field of ViTs for image segmentation. SAM is able to perform zero-shot segmentation, meaning that it can segment objects in images without any prior training on the specific objects. The segmentation happens based on user input, like points, bounding box, or text prompts. The model then returns a set of valid segmentation masks for the input. In most cases, the model outputs up to three segmentation masks that represent the object: the whole, the part, and the subpart. The architecture of SAM and it's working principle is illustrated in Figure 4.7.

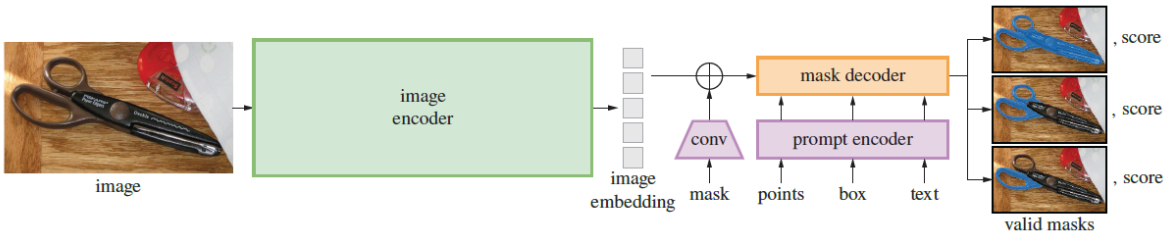


Figure 4.7: SAM architecture and working principle [68].

The model has been trained on a dataset of over 1 billion images and is open-source, making it a promising candidate for the identification of large-scale structures. For example, the model could be used to segment the peaks in the azimuthal and elevation angle histograms. The model would be able to identify the area surrounding the peaks without prior training or resorting to artificially generated training data. The model requires only a prompt as input. In this case, the peak maximum can be provided as a point prompt, which can be identified using basic image filtering techniques like the one provided by the *SciPy* library [70].

4.4. Concluding remarks

In this chapter, we have explored various machine learning techniques that can be applied to the identification of large-scale structures in homogeneous isotropic turbulence. Clustering techniques, such as k-means, M-KNN, DBSCAN, and GMM, provide a way to group data points based on their similarity. However, they often require user-defined parameters, which are usually dependent on the dataset and can be difficult to determine beforehand.

On the other hand, supervised learning techniques, such as CNNs and ViTs, can provide high accuracy in the identification of peaks in the azimuthal and elevation angle histograms. However, CNNs are typically designed to work on Cartesian coordinates, which can introduce distortions in the data when projecting the spherical coordinates onto a 2D plane. To overcome this limitation, cube-maps and HEALPix grids can be used to minimize the effect of the distortion. Additionally, spherical U-Nets can be used to compute the histograms directly on the spherical surface without the need for any projection. However, these methods require a training dataset with labeled data, which is often not available for turbulent flows and would need to rely on artificially generated training data [36], which could introduce assumptions regarding the peak and structure shapes.

Using zero-shot segmentation ViTs like SAM, it is possible to identify the peaks in the azimuthal and

elevation angle histograms without prior training or resorting to artificially generated training data. The model can be used to segment the peaks based on a point prompt, which can be identified using basic image-filtering techniques. This approach allows for a more flexible and adaptable identification process that does not rely on user-defined parameters or assumptions regarding the peak shape. Nevertheless, it would still depend on the bin resolution of the histogram and would require careful considerations for the spherical coordinate system.

Overall, the most suitable method for the identification of large-scale structures in homogeneous isotropic turbulence would be HDBSCAN. Unlike k-means, DBSCAN, or GMM, which require parameters difficult to determine beforehand, HDBSCAN requires only the minimum number of point that compose a cluster. This parameter mainly controls how conservative the clustering is, which in the case of large-scale structures affects how much variation in velocity direction is allowed within a detected structure. Additionally, HDBSCAN can be directly applied to the three velocity components, eliminating the need for corrections related to spherical coordinate systems. In conclusion, HDBSCAN provides a flexible and adaptable clustering technique that can be used to identify large-scale structures in homogeneous isotropic turbulence without the need for complex user-defined parameters or assumptions regarding the peak shape.

5

Methodology

This chapter describes the methodology used to identify large-scale structures in a homogeneous isotropic turbulence velocity field. The methodology consists of three main steps: pre-processing, processing, and post-processing. The pre-processing step is used to prepare the data for clustering, the clustering step is used to identify coherent velocity regions, and the post-processing step is used to extract the structures from the clusters and compute statistics. The flowchart of the identification method is shown Figure 5.1.

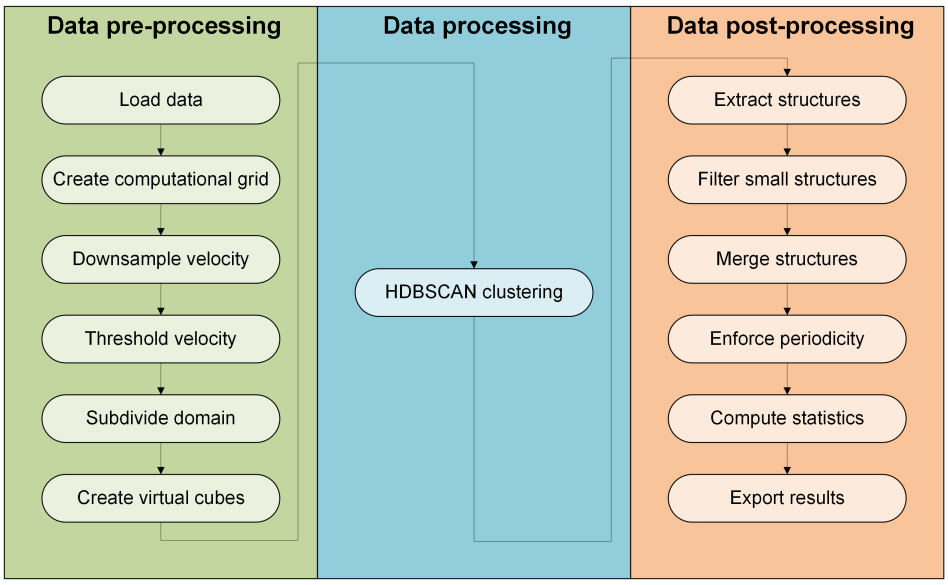


Figure 5.1: Large-scale structure identification methodology flowchart.

5.1. Pre-processing

The first step in the identification process is to preprocess the homogeneous isotropic turbulence dataset. These datasets contain the velocity components at each grid point in the computational domain. The preprocessing procedure is summarized in the flowchart shown in Figure 5.2. The data is first loaded into the program and converted into NumPy arrays [71].

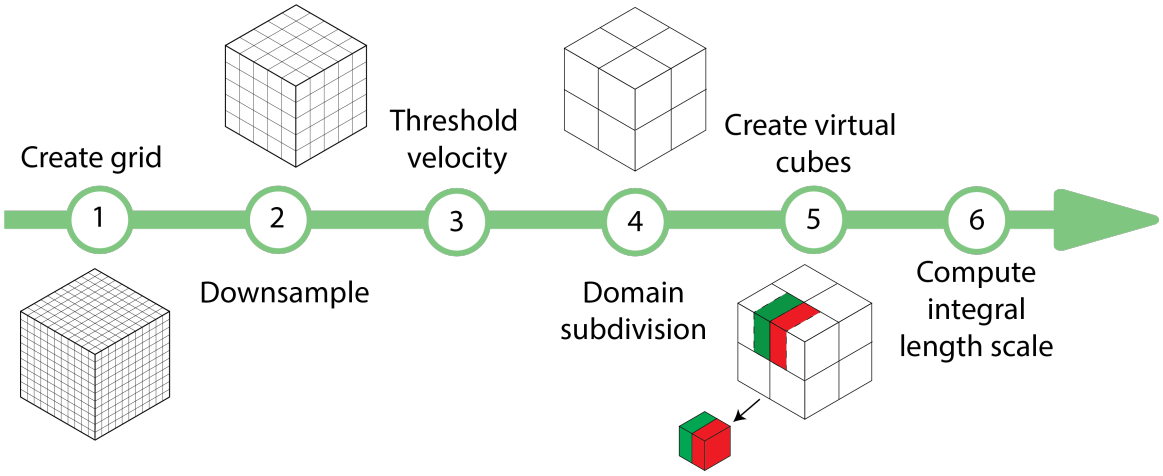


Figure 5.2: Data pre-processing flowchart.

5.1.1. Step 2: Downsampling

Given the large size of DNS datasets, the data is then spatially downsampled to reduce the total number of grid points. The downsampling is performed by selecting every n-th point in each spatial direction, where n is the downsampling factor. For example, $n = 4$, which reduces the size of the dataset by a factor of 4^3, resulting in a substantial reduction in memory usage. Although downsampling reduces the amount of available information, it should not significantly affect large-scale structures, as these typically span multiple grid points. The effect of downsampling on the large-scale structures will be studied in Section 5.6.1.

5.1.2. Step 3: Velocity threshold

Next, a velocity threshold is applied to the velocity field to filter out low-velocity regions and increase the detectability of large-scale structures. The threshold is not expected to affect the identification of large-scale structures, as they are expected to contain most of the kinetic energy of the flow and therefore exhibit higher velocities [30].

The velocity threshold is applied to the velocity magnitude. Any grid points with a velocity magnitude below the specified threshold are excluded from the clustering process.

5.1.3. Step 4: Domain subdivision

Subsequently, the velocity field is subdivided into smaller cubes. This reduces the amount of data that needs to be processed at once, making the processing step less memory-intensive. Furthermore, it increases the detectability of large-scale structures since the presence of large-scale structures makes the subdomains neither isotropic nor homogeneous.

5.1.4. Step 5: Create virtual cubes

The final step in the pre-processing is to create virtual cubes. This is done by creating a set of cubes that overlap with the previous ones. The overlapping cubes created in the interior of the domain are intended to account for structures that span across multiple cubes. Structures located near cube boundaries are expected to appear in both the original and virtual cubes, and can later be merged into a

single structure during the post-processing phase.

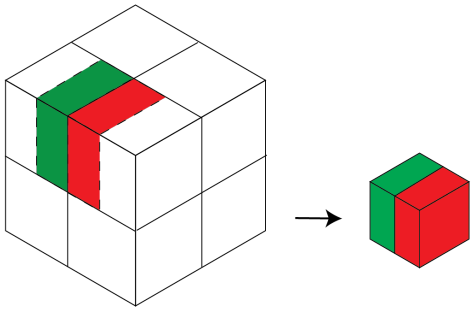


Figure 5.3: Virtual cube creation process.

Virtual cubes are created by taking the first half of the original cube and merging it with the second half of the next cube, as illustrated in Figure 5.3. This process is repeated for all cubes in the domain and in all directions.

5.1.5. Step 6: Integral length scale computation

The turbulent integral length scale is a measure of the largest and most energetic features in a turbulent flow. It represents the characteristic size of the large-scale coherent structures in the flow. The integral length scale is computed using the autocorrelation function of the full downsampled velocity field, as described in [3]. The integral length scale is defined as:

$$L_i = \frac{1}{R_{ii}(0)} \int_0^\infty R_{ii}(r) dr \tag{5.1}$$

where $R_{ii}(r)$ is the autocorrelation function of the velocity field, defined as:

$$R_{ii}(r) = \langle u_i(\mathbf{x}) u_i(\mathbf{x} + r\hat{u}_i) \rangle \tag{5.2}$$

where u_i is the velocity component in the i-th direction, $\mathbf{x}$ is the position vector, $\hat{u}_i$ is the unit vector parallel to the i-th velocity component, r is the separation distance between the two points in space, and $\langle \cdot \rangle$ denotes the ensemble average over the entire domain. The integral length scale is computed for the entire domain in all three spatial directions, resulting in three different length scales L_x, L_y, and L_z. The average integral length scale is then computed as:

$$L = \frac{L_x + L_y + L_z}{3} \tag{5.3}$$

The integral length scale is used to characterize the size of large-scale structures in the flow, as it provides a measure of the characteristic size of the large-scale coherent structures in the flow.

5.2. Processing

The processing step is performed using the HDBSCAN algorithm [46]. This is a density-based clustering algorithm that is well-suited for identifying clusters of varying shapes and sizes within large datasets. HDBSCAN's ability to handle noise and outliers, which can be attributed to small-scale activity in a turbulent flow, makes it particularly useful for identifying large-scale structures. This section will detail the definition of the feature space and the specific parameters used for the algorithm.

5.2.1. Feature space definition

Large-scale structures are expected to show a coherent velocity field, which means that the velocity inside a structure is quasi-uniform in both direction and magnitude. This is in contrast to small-scale structures, which are characterized by high velocity gradients and therefore a rapidly changing velocity field.

The HDBSCAN algorithm can be used to identify points that are close together in the feature space, forming a cluster. The main challenge with this approach is to define the feature space in such a way that it captures the coherent velocity field of large-scale structures. While a similar approach to [10] could be to use the velocity angles defined in spherical coordinates and the points coordinates, this would lead to a very high-dimensional feature space, which would be computationally expensive and difficult to interpret. Additionally, the method would require careful handling of the spherical coordinates, as the angles are periodic and can lead to discontinuities in the feature space.

Instead, the proposed method uses the velocity components in Cartesian coordinates as the feature space. This approach creates a three-dimensional, computationally efficient, and easily interpretable feature space. The points in this space are then clustered using the HDBSCAN algorithm.

An important consideration is that the resulting clusters are not guaranteed to represent a single, spatially continuous structure. The feature space, which excludes spatial coordinates, does not incorporate information about the location of the points. While adding these coordinates could ensure spatial continuity, it would also result in a high-dimensional space that is computationally expensive and physically inconsistent due to the different dimensions of position and velocity. Therefore, spatial continuity will be enforced in the post-processing step. This approach allows for the identification of large-scale structures while maintaining computational efficiency and interpretability.

5.2.2. Definition of the HDBSCAN parameters

A few parameters must be defined before performing clustering with the HDBSCAN algorithm. These include the clustering method, the minimum cluster size, and the minimum sample size, which will be specified in this section.

Clustering method

The HDBSCAN clustering algorithm offers two primary methods for cluster selection: End of Mass (EOM) and leaf. The EOM method identifies clusters by maximizing their excess of mass, a measure that quantifies how long a cluster persists across different density levels. This approach typically yields fewer, larger, and more stable clusters and tends to merge nearby dense regions.

In contrast, the leaf method defines clusters as the terminal nodes ("leafs") of the condensed cluster hierarchy. This results in smaller and more homogeneous clusters. For the analysis of large-scale structures, where a structure's velocity field is expected to be quasi-uniform, the leaf method is more suitable. Using this method ensures that more homogeneous clusters are obtained and prevents the merging of separate high-density regions that could belong to different physical structures.

Minimum cluster size

In HDBSCAN, the minimum cluster size parameter controls the smallest number of points required to form a cluster. A higher value forces the algorithm to find larger, more stable clusters, often merging smaller dense regions into larger ones or labeling them as noise. This effect can be observed in Figure 5.4, where it can be observed that a low minimum cluster size results in many small clusters, while a high minimum cluster size results in fewer and larger clusters.

While it would be intuitive to set the minimum cluster size proportional to the integral length scale to

identify large-scale structures, this approach is complicated by the nature of the feature space. Because the feature space does not ensure spatial continuity of a structure, a single cluster might contain multiple structures or include points that are not part of a coherent structure. As a density-based algorithm, HDBSCAN uses the minimum cluster size as a dynamic density threshold. Therefore, a minimum cluster size based on the integral length scale would have to be adjusted for each density level across domains of varying sizes.

Consequently, it is more effective to set the minimum cluster size proportional to the domain size. This ensures that the clustering remains consistent across datasets with varying sizes, downsampling, resolutions, and domain subdivisions. The minimum cluster size will thus be defined as follows:

$$\text{min\_cluster\_size} = \frac{m}{1000} D \tag{5.4}$$

Where m is a constant that can be adjusted according to the desired level of coherency within a cluster, and D is the detection domain size in grid points. A factor of 1000 is used to simplify the calculations by avoiding very small numbers.

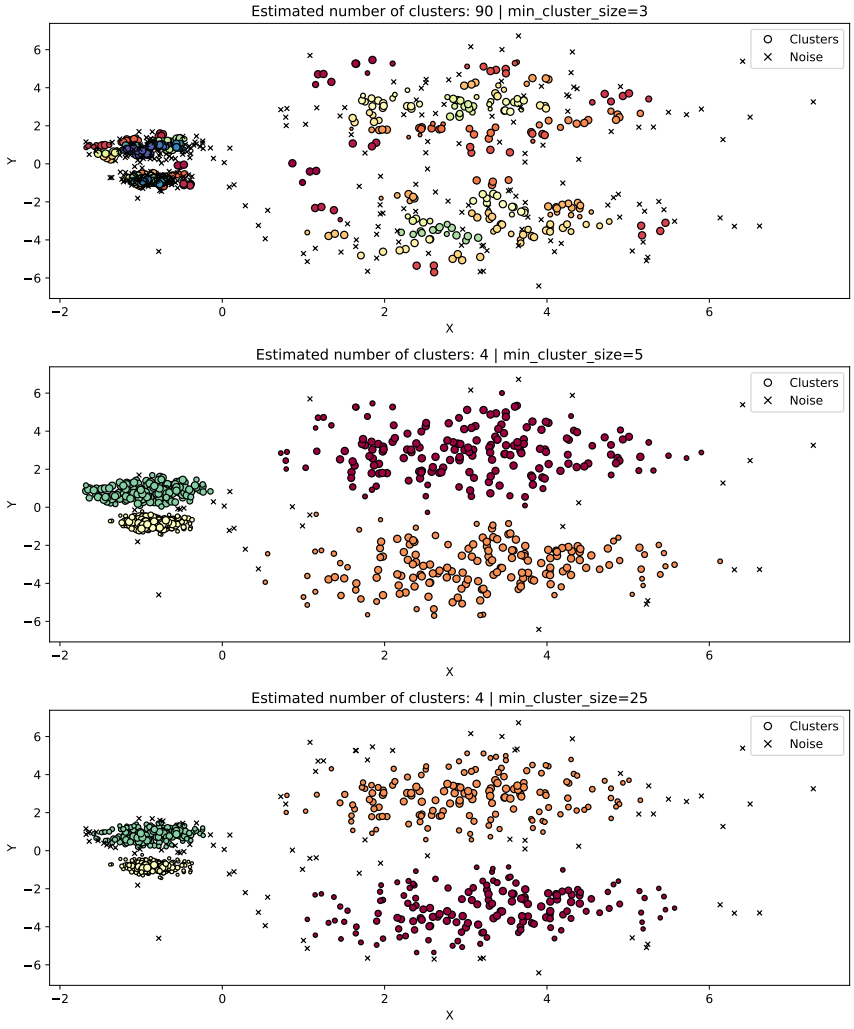


Figure 5.4: Effect of the minimum cluster size on the HDBSCAN algorithm [72]. Subplots present the clustering results for different values of min_cluster_size.

Minimum samples

The minimum sample parameter in HDBSCAN controls the sensitivity of the clustering algorithm to noise in the data. It defines the number of points required to form a core point. Therefore, it also defines a neighborhood size used to estimate local density. A higher value makes the algorithm less sensitive to noise, while a lower value allows more noise in the data. This can be observed in Figure 5.5, where a low minimum samples value results in fewer points being classified as noise, while a high minimum samples value results in more conservative clustering.

The maximum value for the minimum samples parameter is the minimum cluster size, as this ensures that the algorithm does not consider points that are part of a cluster as noise. While a higher minimum samples parameter would reduce the noise in the data, it would also increase the computational cost and memory usage of the algorithm, as a larger neighborhood size must be computed and stored for each point when estimating core distances and constructing the mutual-reachability graph.

The minimum samples will be defined as follows:

$$\text{min\_samples} = \frac{sp}{100}\text{min\_cluster\_size} \tag{5.5}$$

where sp is a constant that can be adjusted according to the expected noise. A small minimum sample parameter is expected to be sufficient for filtering out most of the noise, as a significant part of the points that do not belong to large-scale structures are already filtered by the velocity thresholding applied in Section 5.1.2.

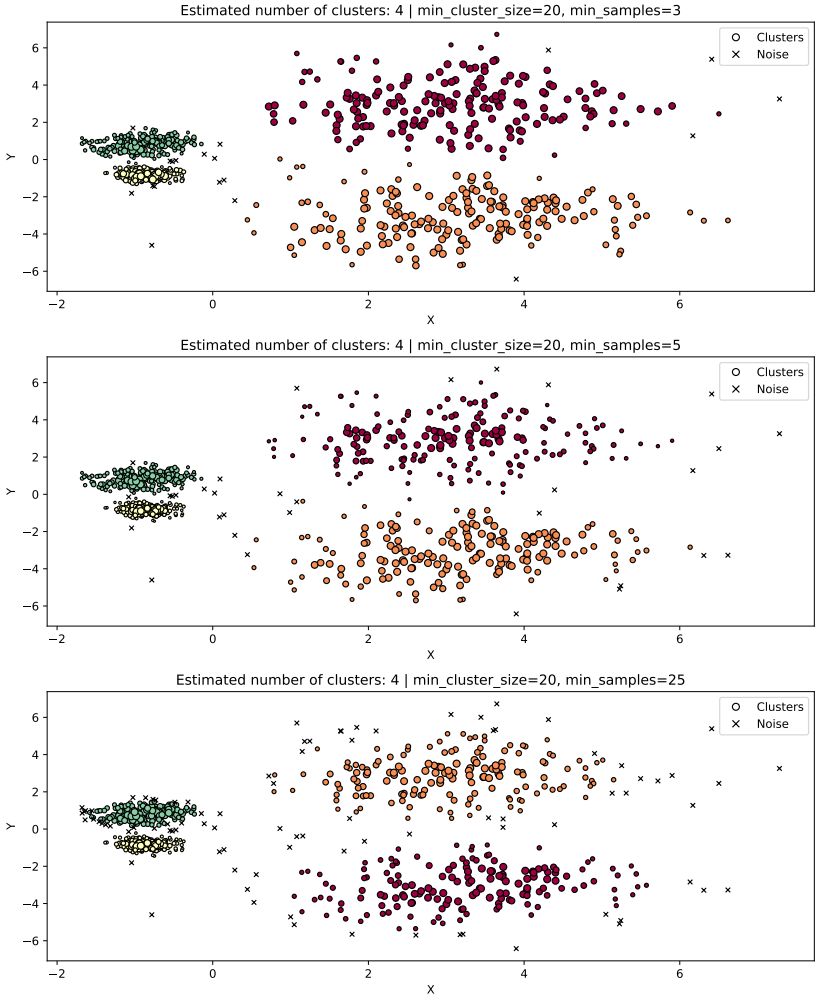


Figure 5.5: Effect of the minimum sample size on the HDBSCAN algorithm [72]. Subplots present the clustering results for different values of min_samples.

5.2.3. Example of clustering results

Figure 5.6 presents an example of clustering results using the HDBSCAN algorithm. The clusters are identified based on the velocity field, with points belonging to the same cluster shown in a single color. Additionally, the clusters are projected into a spherical coordinate system for direct comparison with the methodology of Elsinga and Marusic [10].

The advantage of this approach is that HDBSCAN automatically selects the peak regions in the histogram, even when they are faint and not clearly visible. Furthermore, the use of velocity components as the feature space simplifies the analysis by inherently handling the periodic boundary conditions of the spherical coordinate system. This eliminates the need for any corrective factors near the poles, which were required in previous methods [10, 36].

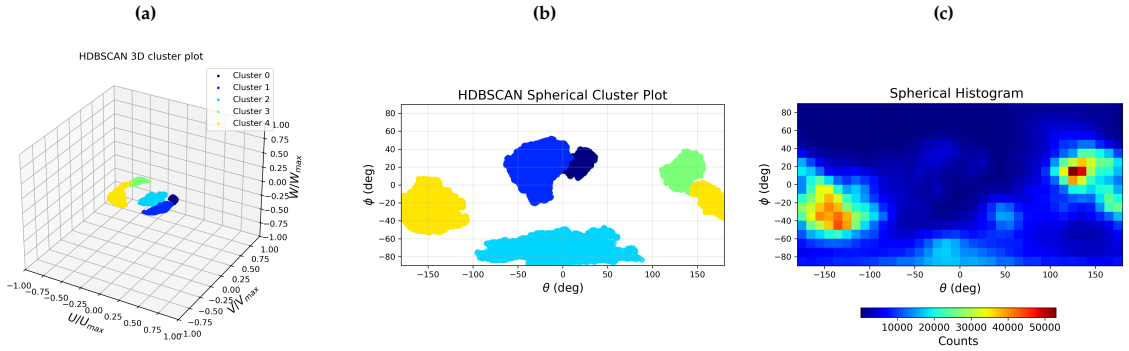


Figure 5.6: Example of clustering results using the HDBSCAN algorithm. **(a)**: 3D representation of the clusters in the feature space **(b)**: 2D projection of the clusters in spherical coordinates. **(c)**: Histogram of the velocity angles in spherical coordinates.

5.2.4. HDBSCAN implementation

The HDBSCAN algorithm is implemented in Python using the hdbscan library [73]. This library provides a multithreaded implementation of the HDBSCAN algorithm, which is suitable for rapidly processing large datasets.

The HDBSCAN algorithm is also available in a GPU-accelerated version, which can significantly speed up the clustering process. The GPU-accelerated version of HDBSCAN is implemented in the RAPIDS library [74]. This implementation is designed to take advantage of the parallel processing capabilities of modern GPUs, significantly improving computational time.

However, the HDBSCAN algorithm is not designed to handle very large datasets, as it requires a significant amount of memory to store the mutual-reachability graph. Therefore, in the scope of this thesis, the GPU-accelerated version of HDBSCAN is limited to datasets up to 256^3 gridpoints after downsampling.

5.3. Post-processing

The post-processing step is performed to extract large-scale structures from the clusters identified by the HDBSCAN algorithm. The post-processing step is summarized in the flowchart shown in Figure 5.7. The post-processing step consists of four main steps: extracting the structures from the clusters, filtering the small structures, merging the structures, and enforcing periodic boundary conditions. Finally, computing statistics of those structures.

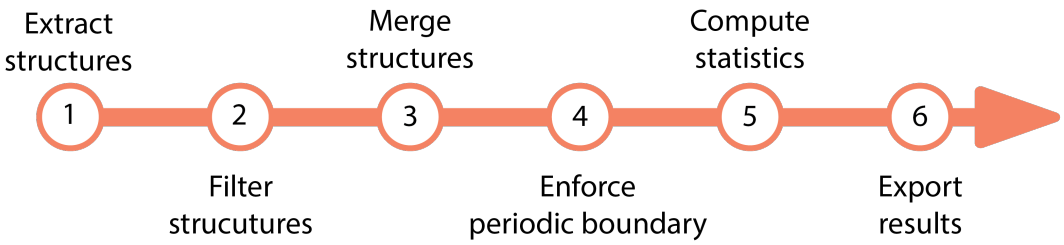


Figure 5.7: Post-processing flowchart.

5.3.1. Step 1: Extracting structures from clusters

The first step in the post-processing is to extract the structures from the clusters identified by the processing step (Section 5.2). Since the HDBSCAN algorithm does not guarantee that the clusters are spatially continuous, the first step is to extract the points that belong to each cluster. This is done by iterating over the clusters and extracting the points that belong to each cluster. Then, for each cluster, the points that belong to that cluster are flagged on the computational grid using an indicator function. This function is defined as follows:

$$I(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in C_i \\ 0 & \text{otherwise} \end{cases} \tag{5.6}$$

where $I(\mathbf{x})$ is the indicator function, $\mathbf{x}$ is the position vector and C_i is the i-th cluster. The indicator function is used to create a binary mask that indicates which points belong to each cluster.

The structures are then extracted from each cluster using the VTK connectivity filter [75]. This filter is used to extract the connected components of the binary mask created by the indicator function. The connectivity filter iterates over the points in the binary mask and performs a breadth-first search [76] to find all the points that are connected to each other. The result is a set of structures, each represented by a set of points marked with the same Structure ID. The connectivity filter is used to ensure that the structures are spatially continuous and represent a single coherent structure.

5.3.2. Step 2: Filtering small structures

The second step in the post-processing is to filter out small structures that are not considered large-scale structures. This is done by computing the volume of each structure and removing those that are smaller than a certain threshold. The threshold is defined as a fraction of the integral length scale, which is computed in Section 5.1.5. The default threshold is set to $0.2L^3$, where L is the integral length scale. This threshold is chosen to ensure that only structures that are large enough to be considered large-scale structures are retained.

5.3.3. Step 3: Merging structures

The third step in the post-processing pipeline is to merge overlapping structures. This is necessary because the full computational domain is subdivided into smaller cubes for analysis, and structures can span across multiple cubes. Given that the virtual cubes created have a 50% overlap, it is expected that

the structures identified in these cubes will overlap with those found in the original cubes.

To ensure that the identified structures are spatially continuous and represent a single coherent entity, the overlapping structures are merged. If a structure overlaps with another by at least 20%, the two are merged into a single structure by performing a boolean union. If the overlap is less than 20%, the structures are considered separate. This process ensures that the identified large-scale structures are not fragmented due to the subdivision of the computational domain.

5.3.4. Step 4: Enforcing periodic boundary conditions

The fourth post-processing step is to enforce periodic boundary conditions. This is accomplished by defining periodic boundary zones, which are region located at the boundaries of the computational domain (Figure 5.8) and are 2% of the domain size in thickness. Each structure that partially enters a periodic boundary zones is checked for continuity across the boundaries. This check is performed by verifying if the structure's portion within a periodic boundary zone overlaps with a part of another structure on the opposing boundary zone. If the overlap is greater than 60% of the volume of the structure in the periodic boundary zones, the two parts are merged into a single coherent structure. This process ensures the continuity of structures across domain boundaries.

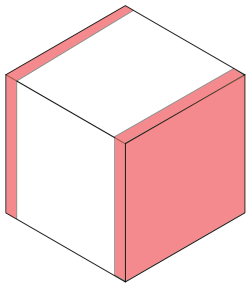


Figure 5.8: Example of periodic boundary zones (in red).

5.3.5. Step 5: Computing statistics

In order to characterize the identified large-scale structures, several statistics are computed. These statistics include the volume percentage, the kinetic energy percentage, and the length scale of each structure. The volume percentage is computed as the ratio of the grid points that belong to the structure ($N_{structure}$) to the total number of grid points in the computational domain (N_{domain}), as shown in Equation (5.7).

$$\text{Volume percentage} = \frac{N_{structure}}{N_{domain}} \times 100\% \tag{5.7}$$

The kinetic energy percentage is computed as the ratio of the kinetic energy of the structure ($E_{structure}$) to the total kinetic energy of the flow (E_{total}), as shown in Equation (5.8). The kinetic energy of a structure is computed as the sum of the kinetic energy of all points that belong to the structure, as shown in Equation (5.9).

$$\text{Kinetic energy percentage} = \frac{E_{structure}}{E_{total}} \times 100\% \tag{5.8}$$

$$E_{structure} = \sum_{i=1}^{N_{structure}} \frac{1}{2}(u_i^2 + v_i^2 + w_i^2) \tag{5.9}$$

The length scale of a structure is computed using Principal Component Analysis (PCA) [77]. PCA is a statistical method that transforms the data into a new coordinate system, where the axes correspond to the directions of maximum variance in the data. The Principal Component Analysis is applied to the

coordinates of the points belonging to a structure. The length scale is then computed for each direction as the maximum distance between the points that belong to the structure in that direction. An example of this computation system is shown in Figure 5.9. Three different length scales (L_1,L_2,L_3) are computed for each structure, corresponding to the three principal directions.

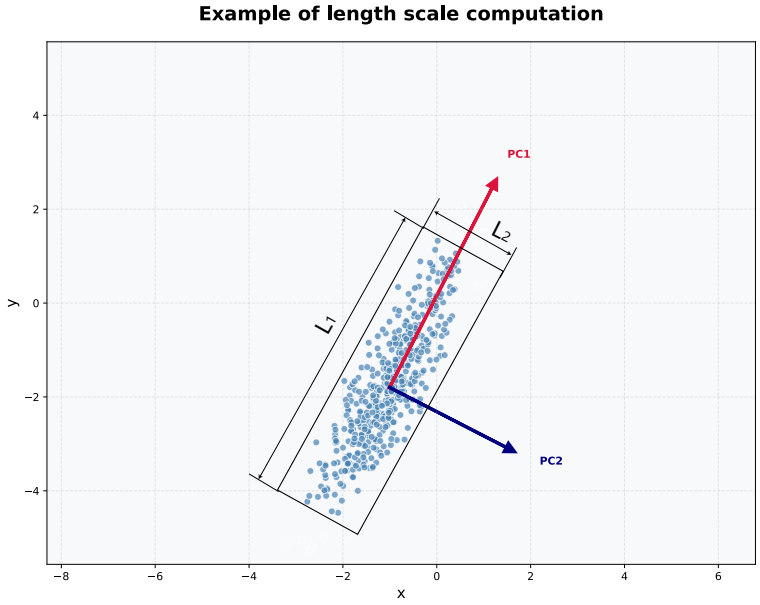


Figure 5.9: Example of length scale computation using PCA.

5.3.6. Step 6: Saving the results

The final step in the post-processing is to save the results. The results are saved in a binary VTK file format. Each structure is assigned a unique structure ID, which is ordered by the volume of the structure, with the largest structure having the lowest ID. Then, each point in the computational domain that belongs to a structure is assigned the unique ID. This allows for easy visualization of the structures in a 3D visualization software. Additionally, the dissipation rate, defined in Equation (5.10), is computed for each point of the domain and saved in the VTK file. This allows for the visualization of the dissipation rate of the flow field, which can provide insights into the energy dissipation mechanisms in the flow.

$$\epsilon = 2\nu S_{ij} S_{ij} \tag{5.10}$$

The statistics computed in Section 5.3.5 are also saved in a separate Comma Separated File (CSV) file, which can be used for further analysis. The statistics include the volume percentage, kinetic energy percentage, and length scale of each structure. Those statistics allow for a detailed analysis of the large-scale structures identified in the flow field.

5.4. Time tracking of large-scale structures

The current method is designed to identify large-scale structures within a single snapshot of the flow field. However, it can be extended to track the temporal evolution of these structures by applying the same clustering and post-processing steps to multiple sequential snapshots.

A key assumption for this approach is that the snapshots are taken at a sufficiently high frequency to ensure that the structures do not change significantly between time steps. This is typically the case for DNS simulations, where the time step is much smaller than the integral timescale.

To track the structures, a matching process is performed by checking for overlap between the structures in consecutive snapshots. If a structure overlaps by at least 60% with a previously identified structure, it is considered to be the same entity and is assigned the same ID. The overlap is defined as the ratio of the

volume of the intersection to the volume of the union of the two structures. Any new structure that does not match a previously identified one is assigned a new ID.

This method also allows for the identification of merging and splitting events. If a structure in the current snapshot overlaps with two or more structures from the previous snapshot, it is identified as a result of a merging event and is assigned the smallest ID of the merged structures. Similarly, if multiple structures in the current snapshot overlap with a single structure from the previous snapshot, they are considered to have originated from a splitting event and are assigned new IDs. This approach could provide valuable insights into the dynamics and interactions of large-scale coherent structures over time.

5.5. Validation case

To assess the proposed method, a validation case is first performed. The case consists of a comparison between the results of the proposed method and the results of a manual extraction performed by Elsinga and Marusic [10]. The validation case uses a forced homogeneous isotropic turbulence dataset provided by Dr. A. A. Wray (CTR 2002, private communication). This data is from a direct numerical simulation with a resolution of 256^3 grid points, a Taylor microscale Reynolds number of $Re_\lambda = 170$, and an integral turbulent length scale of $L = 52$ grid points. The detection is performed on a $2\pi \times 2\pi \times 0.4\pi$ subdomain of the dataset. The velocity threshold is set to 1.3 of the mean velocity magnitude, following the original paper. The results are reported in Figure 5.10. Structures are colored differently to make them easier to distinguish.

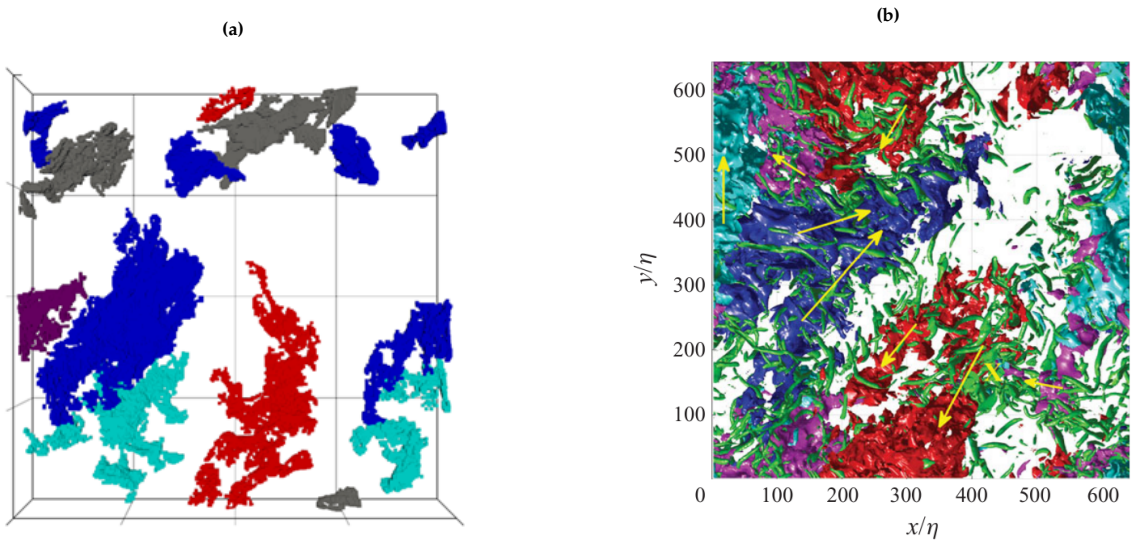


Figure 5.10: Large-scale structures identified in the subvolume of the $Re_\lambda = 127$ dataset. (**a**) Structures extracted using the HDBSCAN algorithm. (**b**) Structures manually extracted by Elsinga and Marusic [10].

The results obtained by HDBSCAN are in good agreement with the results obtained by Elsinga and Marusic [10]. The structures identified by the proposed method are similar to the ones identified manually. It can be noticed that the two largest structures (the central red one and the left blue one) present very similar shapes and positions in both figures. While HDBSCAN is able to identify the larger structures, it does not extract the smaller structures that are present near the subdomain boundaries. However, this can be attributed to the fact that these structures are small and could therefore be filtered out by the post-processing step of the algorithm (Section 5.3). In this case, structures that have a volume smaller than $0.1L^3$ are filtered out. Although a smaller threshold could be employed to extract smaller structures, this would be more time-consuming and is outside the scope of this thesis, which is focused on large-scale structures.

An important remark is that the validation case based on the manual extractions from Elsinga and Marusic [10] cannot be considered a full validation of the method, as the manual extraction is based on a subjective interpretation of the histograms. The manual extraction is performed by visual inspection of the histograms, which can lead to different interpretations by different people. However, it can be considered a good first step to validate the method, as it provides a qualitative comparison between the results obtained by HDBSCAN and the results obtained by manual extraction by an expert in fluid mechanics.

5.6. Stability and robustness of the method

In this section, the effect of different hyperparameters on the identified structures is assessed. The hyperparameters considered are the downsampling factor, the minimum cluster size, the minimum samples, and the velocity threshold. The effect of the variation of these is analyzed on the structures identified in Section 5.5.

5.6.1. Effect of downsampling

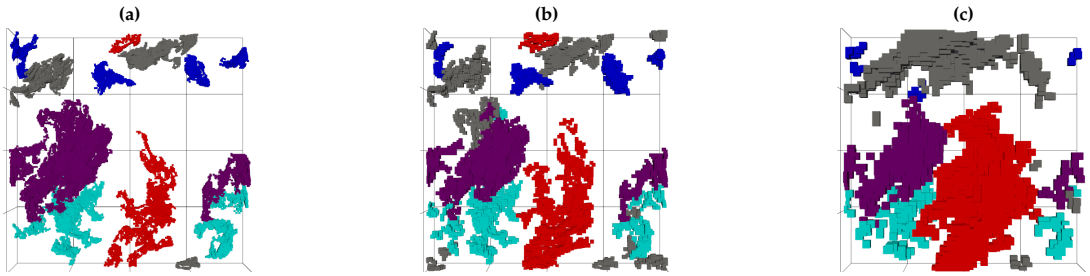


Figure 5.11: Effect of downsampling on the identified structures. (**a**) Downsampling factor $n = 1$ (no downsampling). (**b**) Downsampling factor $n = 2$. (**c**) Downsampling factor $n = 4$.

Figure 5.11 shows the effect of downsampling the dataset on the identified structures. The downsampling factor n is applied in all directions to the original dataset, resulting in a reduction in the number of grid points by a factor of n^3. The results show that the method is robust to downsampling, as the same structures are identified in all cases, regardless of the downsampling factor. The structures increase in size with downsampling, which is likely due to the fact that smaller structures are more likely to be merged because the gaps between them could be removed, making the clusters more continuous.

While an appropriate amount of downsampling does not significantly affect the results, it can lead to a significant reduction in memory usage and processing time. In this dataset, the Reynolds number is $Re_\lambda \approx 170$, which is relatively low, and the number of grid points in the domain is not too large. Therefore, excessive downsampling reduces the spatial resolution too far for meaningful results. For example, in the $n = 4$ case shown in Figure 5.11c, the integral length scale spans only $L = 14$ grid points, which are too few to resolve the large-scale flow structures accurately. While the same structures are identified, their shape and size are significantly altered, and the results become less reliable. Nevertheless, for larger datasets, such as the $Re = 1131$ case, where a higher number of grid points is available, a higher downsampling factor can be used with limited information loss, as there should still be sufficient grid points within an integral length scale.

5.6.2. Effect of velocity threshold

Figure 5.12: Effect of velocity threshold on the identified structures. (**a**) Velocity threshold $v = 1.0\langle|u|\rangle$. (**b**) Velocity threshold $v = 1.3\langle|u|\rangle \approx \langle|u|\rangle + 0.7\sigma_{|u|}$.

Decreasing the velocity threshold results in larger structures and a decrease in HDBSCAN's clustering accuracy. This is because a lower threshold includes more points that are not part of any large-scale structure in the clustering process. The effect of the velocity threshold on the identified structures is shown in Figure 5.12. The results show that the same structures are identified in all cases, but their size and shape change significantly with the velocity threshold. As the velocity threshold decreases, the structures become larger and more diffuse as more points are included in the clusters.

HDBSCAN is a density-based clustering algorithm, which means that it is sensitive to the density of points in the data. While HDBSCAN is robust to noise in the data, for larger clusters, the minimum samples parameter plays a crucial role in determining the accuracy of the clustering in noisy data. Since a smaller minimum samples size considers fewer points in the neighborhood, it is likely that multiple small clusters are detected in noisy data. Those clusters are then merged into larger clusters, which leads to larger structures being identified. This is particularly evident in the $v = 1.0\langle|u|\rangle$ case shown in Figure 5.12a, where the structures are significantly larger than in the $v = 1.3\langle|u|\rangle$ case. While it would be possible to increase the minimum samples parameter to reduce the size of the structures, this would lead to a significant increase in computational cost and memory usage, as the algorithm would need to compute a larger neighborhood for each point. Following the methodology of Elsinga and Marusic. [10], a velocity threshold $v = 1.3$ times the mean velocity magnitude will be used in this thesis. This choice is based on the premise that large-scale structures contain most of the flow's kinetic energy [30].

5.6.3. Effect HDBSCAN hyperparameters

This section discusses the effect of the HDBSCAN hyperparameters on the identified structures. The hyperparameters considered are the minimum cluster size and minimum samples. The effect of these parameters is analyzed by varying them and observing the changes in the identified structures.

Effect of minimum cluster size

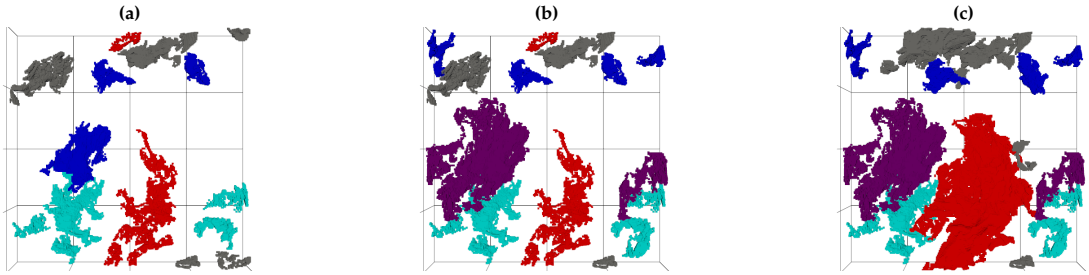


Figure 5.13: Effect of minimum cluster size on the identified structures. (**a**) Minimum cluster size $m = 1.5$. (**b**) Minimum cluster size $m = 3$. (**c**) Minimum cluster size $m = 6$.

The minimum cluster size is a hyperparameter of the HDBSCAN algorithm that specifies the smallest number of points required to constitute a cluster. When identifying large-scale structures, it regulates the degree of coherency allowed in the velocity field within a cluster. The effect of this parameter is shown in Figure 5.13. The results indicate that it influences the size of the structures identified but does not have any significant impact on their overall number, positions, or shapes. As anticipated, increasing the minimum cluster size expands the cluster region, which allows higher variation in the velocity direction inside a cluster and results in larger structures.

Because there is no widely accepted definition of what constitutes a large-scale structure, a minimum cluster size of $m = 3$ will be used for future analysis. This value is chosen based on the observation that the velocity field inside a cluster is coherent enough to be considered a large-scale structure, while still allowing for some variation in the velocity direction. Additionally, it has been observed that the minimum cluster size behaves consistently across all datasets when scaled with the domain size, meaning that the same minimum cluster size can be used for different datasets. The supporting results and behavior in larger datasets are shown in Chapter 7.

Effect of minimum samples

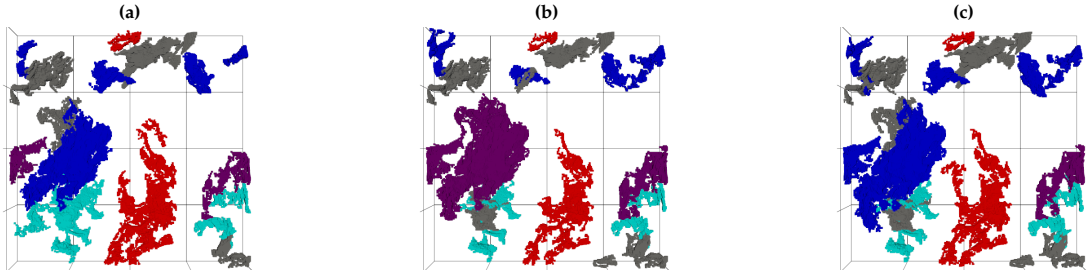


Figure 5.14: Effect of minimum samples on the identified structures. (**a**) Minimum samples $sp = 0.5$. (**b**) Minimum samples $sp = 1$. (**c**) Minimum samples $sp = 2$.

The minimum samples parameter is another hyperparameter of the HDBSCAN algorithm that controls the neighborhood size used to estimate local density. It is used to control the sensitivity of the clustering algorithm to noise in the data. The effect of the minimum samples parameter on the identified structures is shown in Figure 5.14. The results show that the minimum samples parameter does not have any significant effect on the identified structures, as the same structures are identified in all cases. While this holds true at high velocity thresholds, at lower thresholds the noise from the small-scale increases, which might make the clustering more sensitive to the minimum samples parameter.

Increasing the minimum samples parameter raises both computational cost and memory usage, since a larger neighborhood size must be computed and stored for each point when estimating core distances and constructing the mutual-reachability graph. In order to increase the algorithm's efficiency, a minimum samples of $sp = 2$ is used. This setting provides some noise suppression while leaving the identified structures essentially unchanged.

5.6.4. Selected hyperparameters for the identification of large-scale structures

The results of the previous sections show that the proposed method is robust to downsampling, minimum cluster size, and minimum samples. The velocity threshold has a significant effect on the size of the identified structures but does not affect the overall number of structures identified or their position. Based on these results, the parameters reported in Table 5.1 are selected for the detection of large-scale structures in this thesis. These parameters are chosen to balance the computational cost and memory usage of the algorithm while still providing accurate results. The downsampling factor will be set on a case-by-case basis, depending on the size of the resolution of the dataset and the available computational resources, ensuring that enough grid points are contained within an integral length scale.

Parameters	**Value**
Minimum cluster size factor (m)	3
Minimum samples factor (sp)	1
Velocity thresholds	$1.3 \langle \lvert u \rvert \rangle$

Table 5.1: HDBSCAN parameters used for the detection of large-scale structures.

6

Datasets

This chapter describes the datasets used in this study for the identification of large-scale structures in homogeneous isotropic turbulence. Each dataset contains three-dimensional velocity fields obtained from Direct Numerical Simulations (DNS).

6.1. Datasets Description

This thesis uses Direct Numerical Simulation (DNS) data of homogeneous isotropic turbulence. All datasets are set in a periodic cubic domain of size 2π and use a uniform grid. A summary of these datasets, which span a wide range of Taylor-microscale Reynolds numbers ($Re_\lambda \approx 37-1131$), is provided in Table 6.1.

This thesis study includes both forced and decaying turbulence cases. The lower Reynolds number datasets are decaying, while the higher Reynolds number ones are sustained by spectral forcing. Specifically, the $Re_\lambda = 433$ dataset injects energy at low wavenumbers ($|k| < 2$), and the $Re_\lambda = 730$ and $Re_\lambda = 1131$ datasets use a similar approach with energy injected for $|k| < 2.5$. The analysis in Sections 7.1 and 7.2 relies only on single snapshots of the velocity field, and no time analysis will be conducted in those sections. An example of such a snapshot, visualizing several key flow quantities, is shown in Figure 6.1.

It is important to note that the datasets have different resolutions, defined as the number of grid points per integral length scale. The lower Reynolds number datasets have a lower resolution, while the higher Reynolds number datasets have a higher one. This difference is expected to affect the number of structures detected. To preserve the resolution of these structures, the lower Reynolds number datasets will be analyzed without downsampling. For each dataset, the downsampling factor d is reported in Table 6.1.

$\mathbf{Re}_\lambda$	N	$2\pi/L$	d	Forced or Decaying	Reference	Institution
37.1	128	7.52	1	D	[78]	Tokyo Tech
64.9	128	6.73	1	D	[78]	Tokyo Tech
97.1	256	8.53	1	D	[78]	Tokyo Tech
141.1	400	7.69	1	D	[78]	Tokyo Tech
222	640	6.15	2	D	[78]	Tokyo Tech
393	1536	7.14	4	D	[78]	Tokyo Tech
433	1024	4.65	2	F	[79]	Johns Hopkins University
730	2048	5.12	4	F	[80]	Nagoya University
1131	4096	5.73	8	F	[80]	Nagoya University

Table 6.1: Overview of the datasets used in this study. The datasets are characterized by their Reynolds number Re_λ, number of grid points N per dimension (in the 2π periodic box), number of integral length scales per dimension $2\pi/L$, downsampling factor used d, and whether they are forced or decaying.

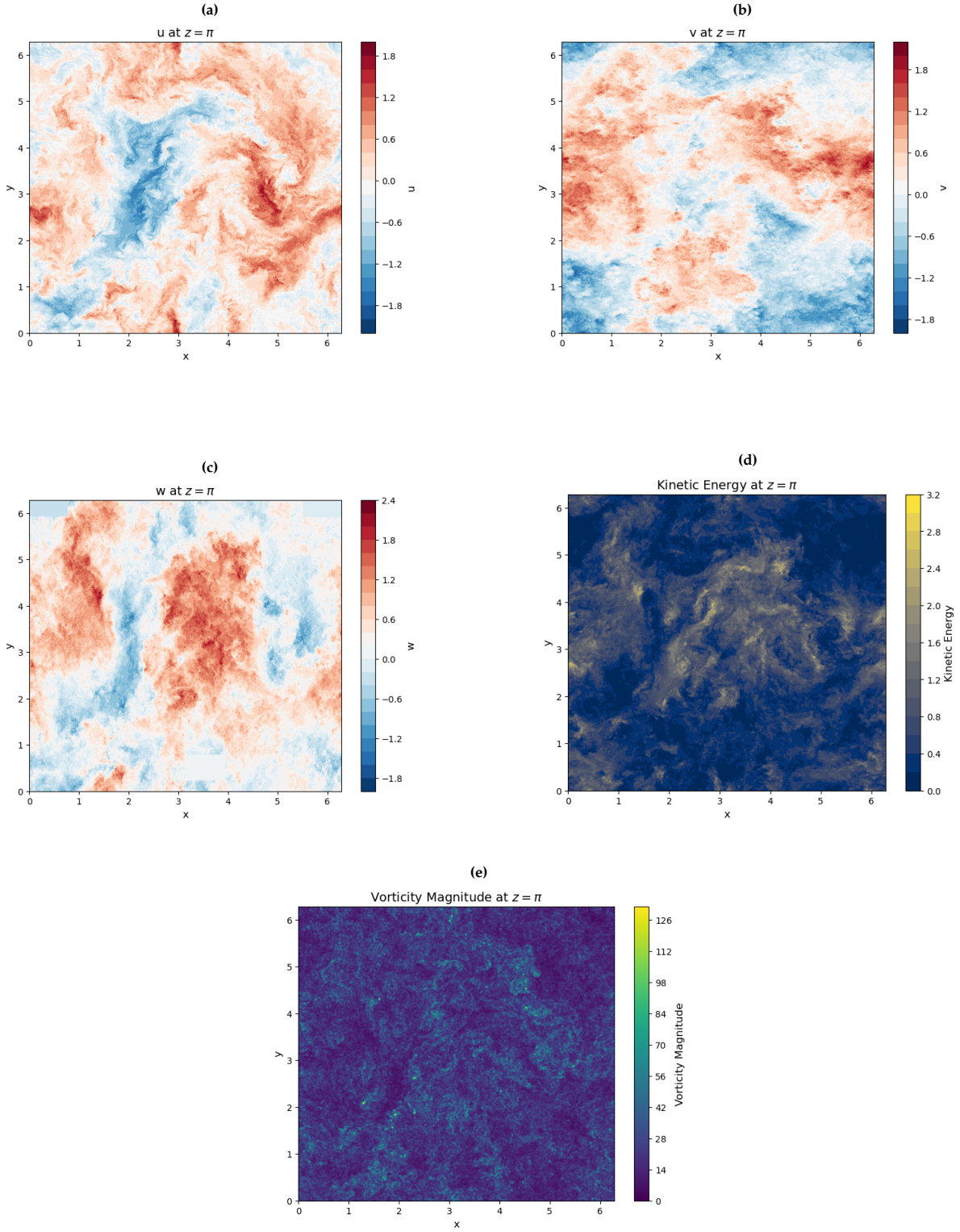


Figure 6.1: Visualization of a single snapshot from the $Re_\lambda = 1131$ dataset [14]. All quantities are shown on the two-dimensional cross-sectional plane at $z = \pi$. **(a)** the u velocity component. **(b)** the v velocity component. **(c)** the w velocity component. **(d)** the kinetic energy field. **(e)** the vorticity magnitude field.

A time-resolved dataset is also used in Section 7.3. The simulation uses a time step of $\Delta t_{\text{sim}} = 2 \times 10^{-4}$ s and runs for a total of $T = 10.056$ s. Snapshots are written every $\delta t = 2 \times 10^{-3}$ s (every 10 simulation steps), resulting in 5028 time samples. The dataset is characterized by $Re_\lambda = 433$ and a large eddy turnover time of $T_L \approx 1.99$. The forcing is applied using a spectral forcing term that injects energy into low wave numbers $|k| < 2$. The dataset is described in more detail in Table 6.2.

$\mathbf{Re}_\lambda$	**N**	$\mathbf{2\pi/L}$	$\mathbf{\Delta t_{sim}}$ **[s]**	$\mathbf{T_L}$ **[s]**	**d**	**Reference**	**Institution**
433	1024	4.6	0.0002	1.99	4	[79]	Johns Hopkins University

Table 6.2: Overview of the time-resolved dataset used in this study. The dataset is characterized by its Reynolds number Re_λ, number of grid points N per dimension (in the 2π periodic box), time averaged number of length scales per dimension $2\pi/L$, simulation time-step Δt_{sim}, large eddy turnover time T_L, and downsampling factor$d$.

7

Results

This chapter presents the results obtained by applying the identification method described in the previous chapters. The results are organized into three sections. The first section focuses on a single representative dataset to illustrate the main features of the identified structures. The second section examines the results for different Reynolds numbers and discusses the scaling behavior of these structures with Reynolds number. Finally, the third section analyzes a time-resolved dataset to capture the temporal evolution of the identified structures.

7.1. Detailed analysis of the $\mathbf{Re}_\lambda = 1131$ case

In this section, we analyze a single dataset to demonstrate the effectiveness of the identification method. The chosen dataset is the Nagoya dataset at $Re_\lambda = 1131$ as it has the highest Reynolds number among the available datasets, making it more representative of real flows. It is also the largest dataset, which allows an assessment of the computational performance of the method.

7.1.1. Identified structures

Figure 7.1 shows all the identified structures in the Nagoya dataset at $Re_\lambda = 1131$ in the entire $2\pi \times 2\pi \times 2\pi$ box. The structures are visualized as volumes within the flow field and are randomly colored to make them easier to distinguish. It can be noticed that the identified structures display a variety of shapes and sizes, and they occupy a significant portion of the domain. From this visualization, it is evident that the structures are not isolated, but rather located in close proximity to each other. This suggests that interactions among these structures might have a significant influence over the turbulent dynamics and the behavior of smaller-scale features in the flow.

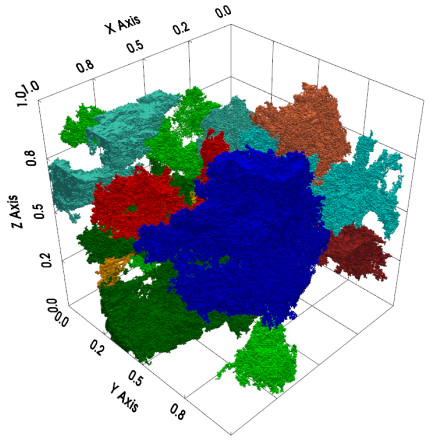


Figure 7.1: Identified structures in the Nagoya dataset at $Re_\lambda = 1131$.

Figures 7.2 and 7.3 provide two representative examples of identified structures. In each of these figures, three elements are shown:

(i) The spherical angles' histogram probability density function of the identified structure, as used in [10] which indicates coherence in the velocity orientation within the structure.

(ii) The volume representation of the structure within the computational domain, showing the spatial extent of the identified structure.

(iii) The vector representation of the velocity field inside the identified structure, visualized using glyphs.

As can be seen in Figures 7.2a and 7.3a, each structure produces only a single peak in the spherical-angle probability density function. Therefore, the identified structures are consistent with the histogram-based identification method (Section 3.6). Additionally, from Figures 7.2c and 7.3c it can be observed that within both structures the velocity field is quasi-uniform, with velocity vectors oriented consistently in nearly the same direction. This confirms that the structures identified by our method are indeed coherent flow regions where the velocity field exhibits organized, quasi-uniform behavior, further validating the detection method.

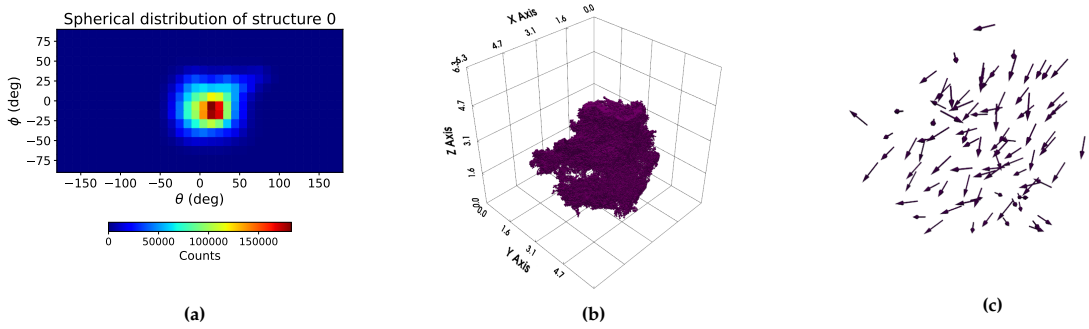


Figure 7.2: Example of detected structure in the $Re_\lambda = 1131$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

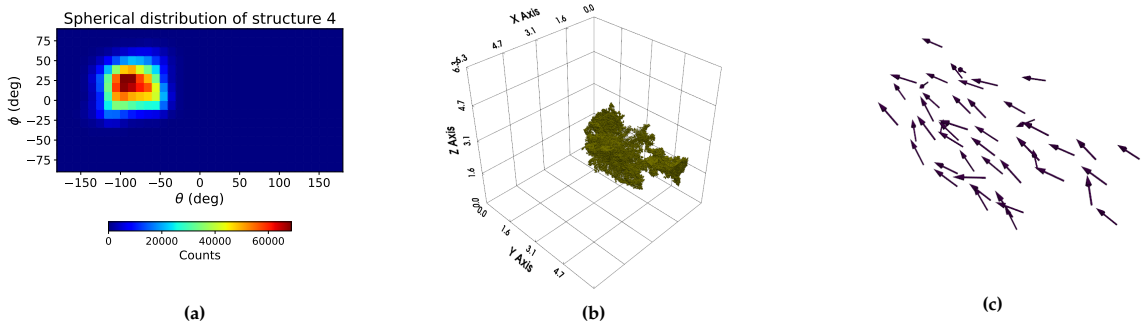


Figure 7.3: Example of detected structure in the $Re_\lambda = 1131$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

7.1.2. Length scale, volume and kinetic energy of identified structures

Figure 7.4 shows the length scales of the identified structures computed using PCA. All the identified structures show a length scale of the order of the integral length scale of the flow field, which is approximately 0.2 times the computational domain. Additionally, as can be seen in Figure 7.5, the structures are energy carrying. This means that the structures contain a higher percentage of the flow's kinetic energy than the volume they occupy. While this may be partly due to the velocity threshold used in the identification algorithm, it can still be interpreted as consistent with classical turbulence theory, where energy is predominantly stored in the large scales [30]. These findings indicate that the identified

coherent structures are large-scale features of the flow, and therefore can be classified as large-scale structures.

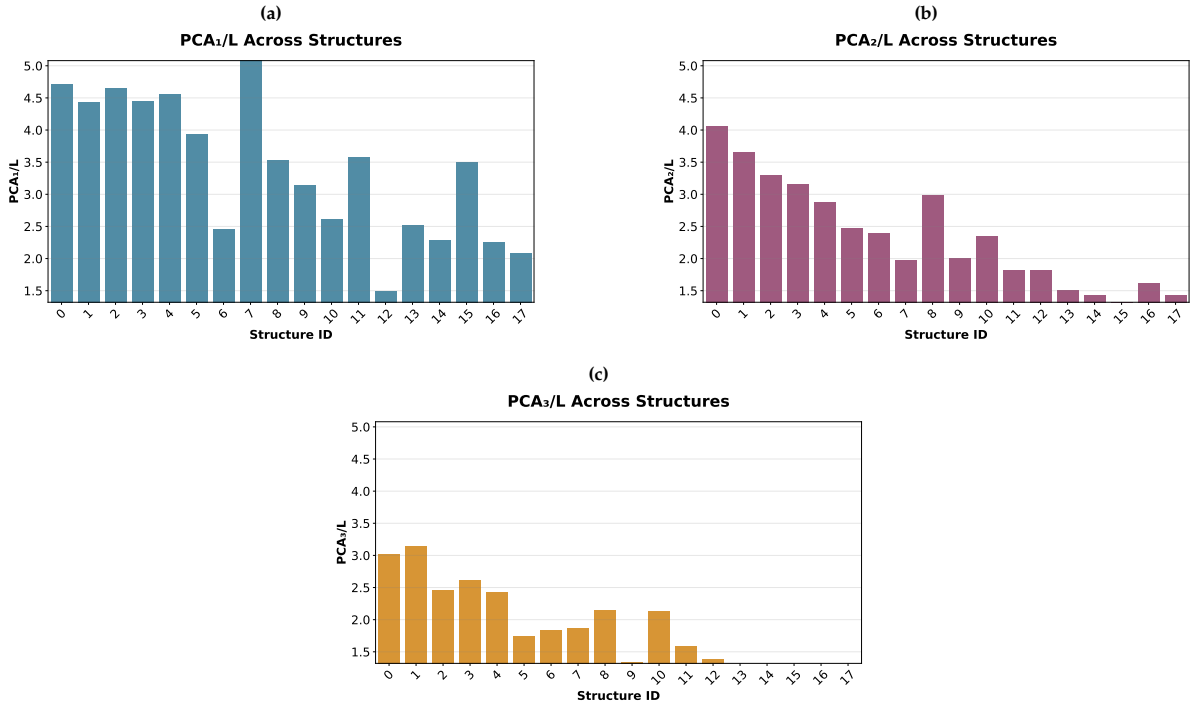


Figure 7.4: PCA length scales distribution of each identified structure at $Re_\lambda = 1131$. **(a)** First principal component. **(b)** Second principal component. **(c)** Third principal component.

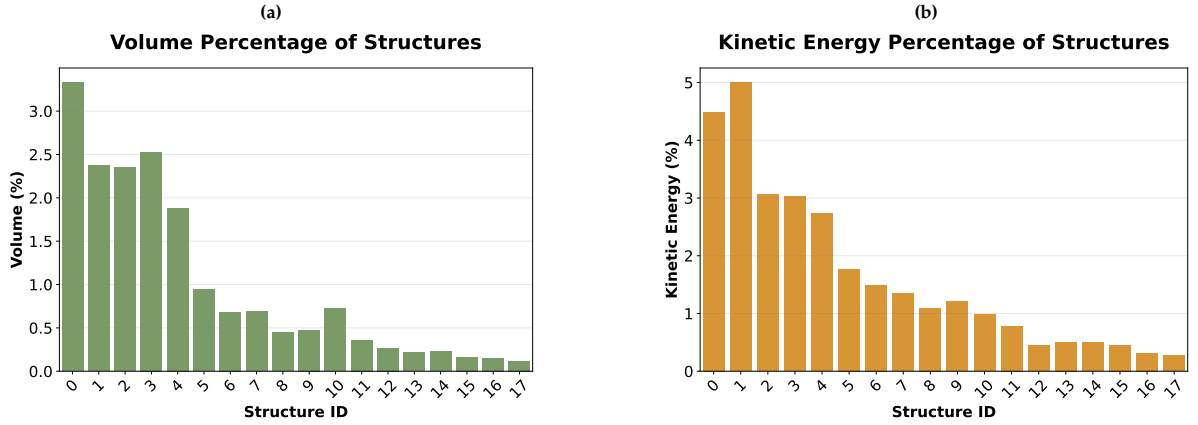


Figure 7.5: Percentage contribution of each structure to the full flow field. **(a)** Volume. **(b)** Kinetic energy.

Figure 7.6 presents the distribution of the ratios of the principal components of the identified structures. A ratio of 1 indicates that the structure is isotropic in the direction of the two PCA components, while a ratio greater than 1 indicates that the structure is elongated. It can be observed that the majority of the identified structures have at least one ratio greater than 1. This suggests that the identified structures are predominantly anisotropic in nature.

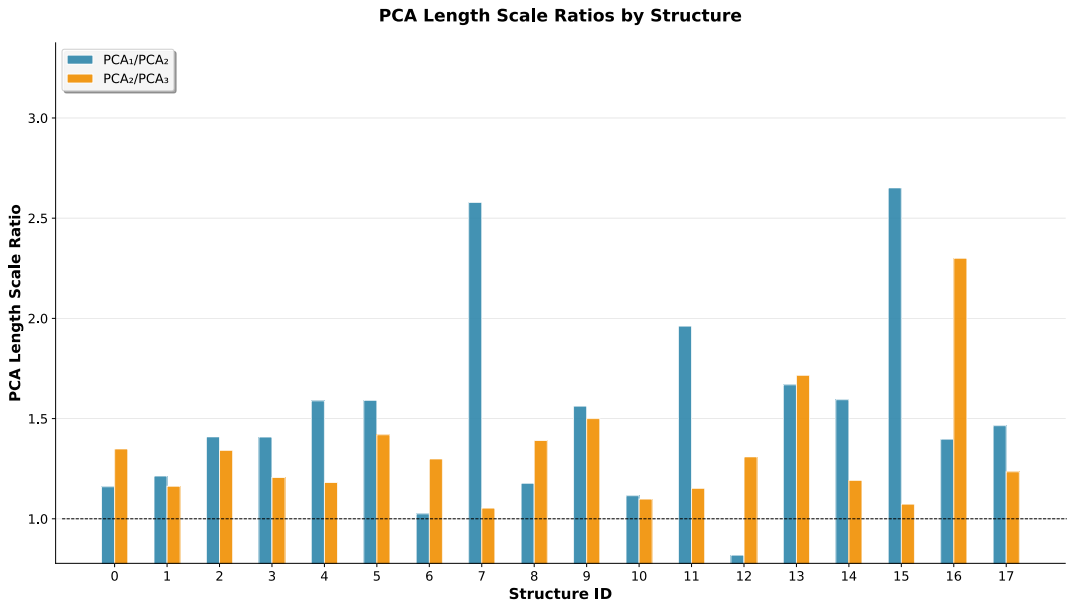


Figure 7.6: PCA ratios of identified structures in the Nagoya dataset at $Re_\lambda = 1131$.

7.1.3. Interaction between structures

Recent studies have theorized that the interaction between large-scale structures can lead to high dissipation rates regions in the flow between those structures [10, 81]. These studies suggest that the motion of two large scale structures can lead to high shear regions in the flow, which can lead to high dissipation rates. In this section, we will analyze the interaction between two large-scale structures in the Nagoya dataset at $Re_\lambda = 1131$ and its effect on the dissipation rate. It is important to remark that the dissipation rate is defined based on the velocity gradients, which are computed using a second order finite difference scheme. Since the dataset is downsampled to reduce the computational cost, the gradients are not as accurate as they would be in a higher resolution dataset. Therefore, the results and the analysis presented in this section should only be considered as qualitative and represent a coarse-grained dissipation rate.

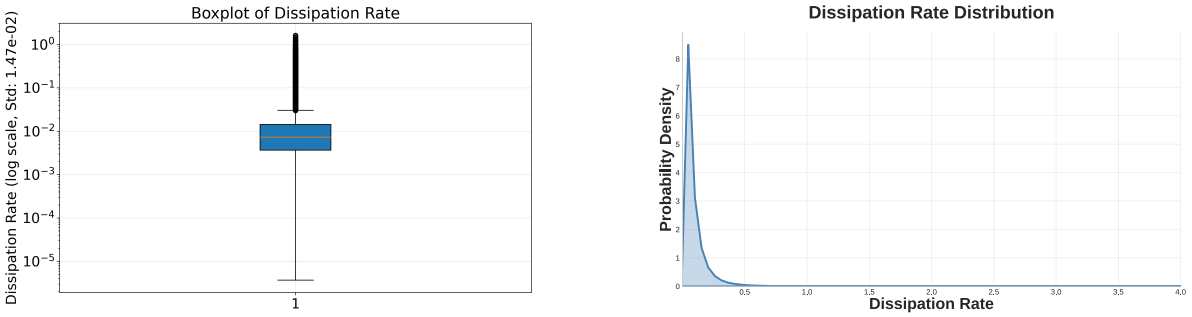
The probability density function of the dissipation rate and the box-plot in the Nagoya dataset at $Re_\lambda = 1131$ are shown in Figure 7.7. As shown in the box-plot, the dissipation rate in the Nagoya dataset at $Re_\lambda = 1131$ is characterized by a significant upper tail, with many values falling beyond the upper whisker. This is indicative of intermittency [30] and points to regions of exceptionally high dissipation. The substantially higher dissipation rates in these outlier regions, compared to the mean, suggest that these areas account for a significant portion of the total dissipation in the flow.

Figure 7.8 illustrates the interaction between two large-scale structures in the Nagoya dataset at $Re_\lambda = 1131$. In Figure 7.8a, contours of the dissipation rate at $< \epsilon > +5\sigma_\epsilon$ highlight intermittent events. Figure 7.8b highlights the presence of large-scale structures located within the high-dissipation region. Figure 7.8c further illustrates that these high-dissipation regions lie between the two structures. Additionally, the velocity field in Figure 7.8d exhibits counter-directed vectors across the gap, consistent with a shear layer forming between the structures. This trend is corroborated by Figure 7.9, which plots the dissipation rate along the line connecting the structure centroids and shows significantly higher values between them. Together, these observations suggest that interactions between neighboring large-scale structures induce strong shear between them, leading to elevated dissipation regions, in agreement with previous theoretical expectations and studies [10, 81].

Furthermore, recent studies by Park and Lozano-Durán [11] have shown that high energy transfer events and high dissipation rates are located in between two hairpin-like high enstrophy regions. The detected large scale structures feature a quasi-uniform velocity field, which means that the enstrophy inside the structures is low. However, a rapid change in the velocity direction while moving from one structure to the other can result in high enstrophy regions. Therefore, while the structures themselves do not

contain high enstrophy, the interaction between them can lead to high enstrophy regions in the flow.

In conclusion, the interaction between large-scale structures in turbulent flows can lead to significant shear layers and high dissipation rates. These findings align with theoretical analyses and previous studies, highlighting the importance of understanding the dynamics of large-scale structures in homogeneous isotropic turbulence.



(a) Box-plot of dissipation rate.

(b) Probability density function of dissipation rate.

Figure 7.7: Dissipation rate distribution in the Nagoya dataset at $Re_\lambda = 1131$. **(a)** Box-plot. **(b)** Probability density function.

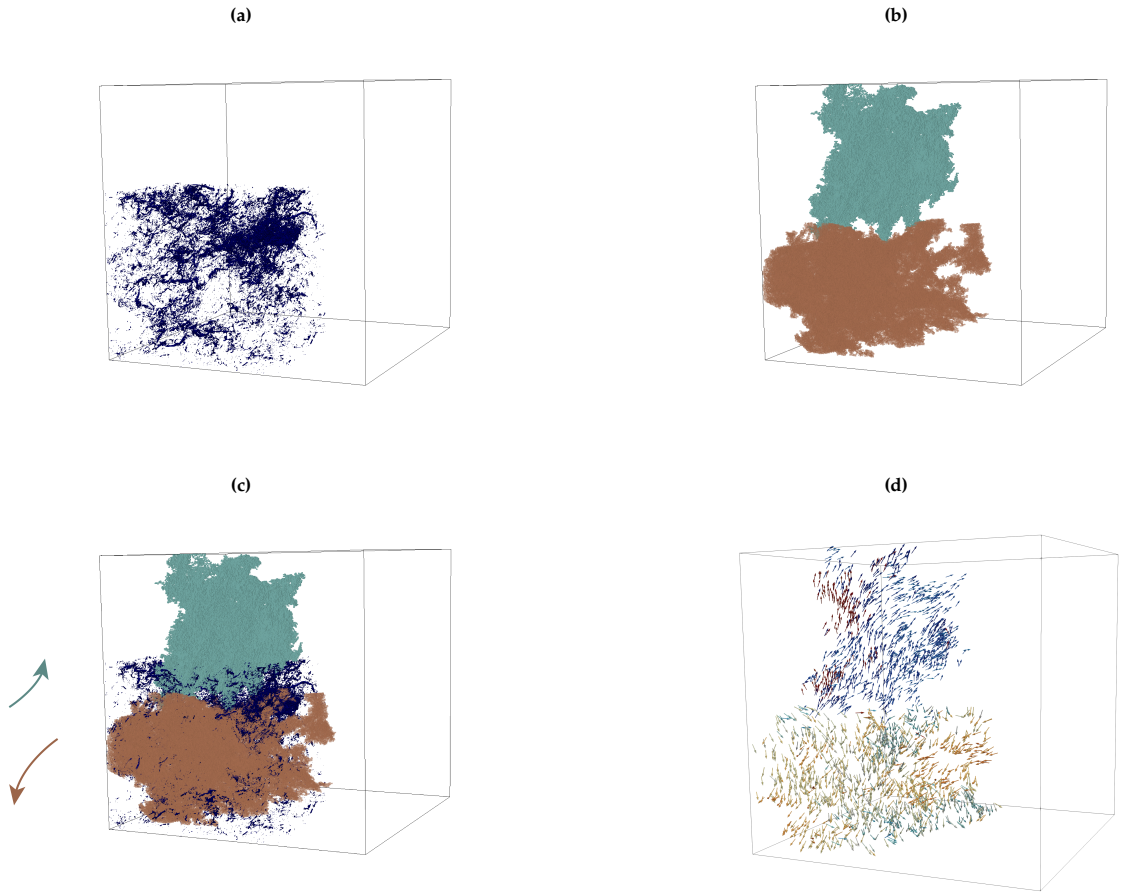


Figure 7.8: Interaction between structures in the Nagoya dataset at $Re_\lambda = 1131$. **(a)** Contours of dissipation rate at $< \epsilon > +5\sigma_\epsilon$. **(b)** Large scale structures. **(c)** Detail of the interaction between structures. **(d)** Velocity field of the structures.

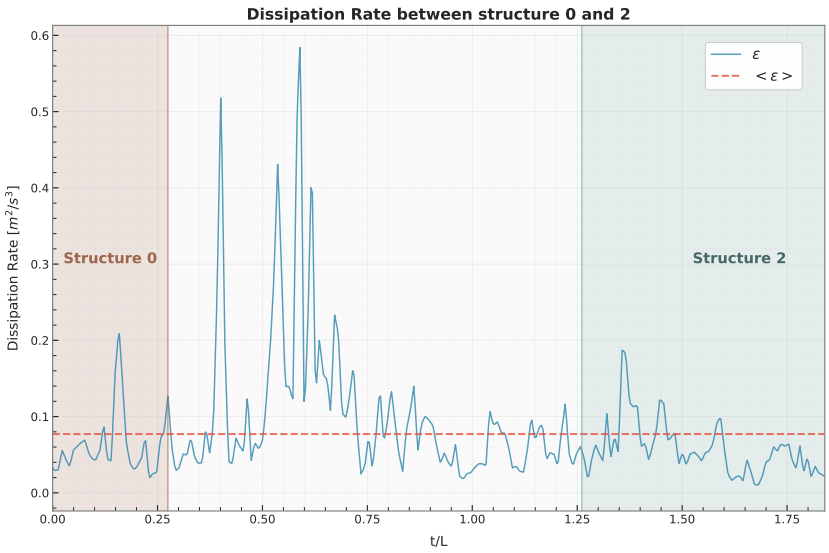


Figure 7.9: Dissipation rate in the Nagoya dataset at $Re_\lambda = 1131$ along the line joining the centers of the two large-scale structures. t is the distance from the largest detected structure to the third-largest one.

7.1.4. Computational performance

The clustering process is computationally efficient, taking approximately one hour to complete on an *Intel Xeon w7-2475X* with a peak memory usage of 25 GB. The post-processing step also required one hour, with a peak usage of 75 GB due to the large number of variables computed (e.g. dissipation rate, kinetic energy) for the analysis. While this memory usage could be optimized by saving intermediate results to disk, this is considered outside the scope of this thesis and is suggested for future work. Overall, the method is computationally efficient and suitable for large datasets.

7.2. Effect of Reynolds number

The current section analyzes the effect of Reynolds number on the identified structures. The analysis is performed on a set of datasets with different Reynolds numbers, ranging from $Re_\lambda = 37$ to $Re_\lambda = 1131$. The datasets are summarized in Table 6.1. The analysis focuses on the scaling behavior of the identified structures. The results are presented in terms of the length scales of the identified structures and their energy contribution.

7.2.1. Identified structures

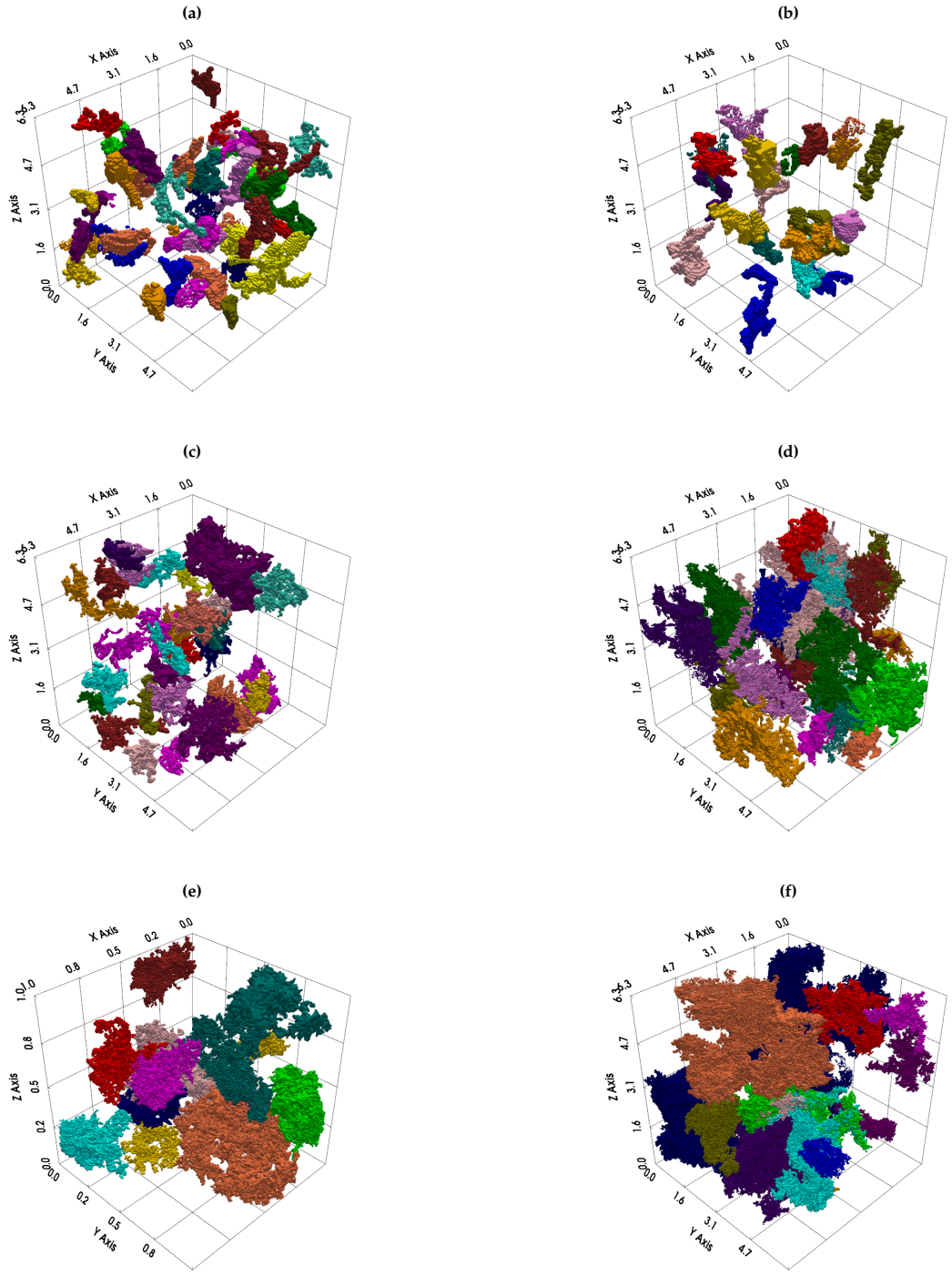


Figure 7.10: Identified structures at different Reynolds numbers (Decaying cases). **(a)** $Re_\lambda = 37.1$. **(b)** $Re_\lambda = 64.9$. **(c)** $Re_\lambda = 97.1$. **(d)** $Re_\lambda = 141.1$. **(e)** $Re_\lambda = 222$. **(f)** $Re_\lambda = 393$.

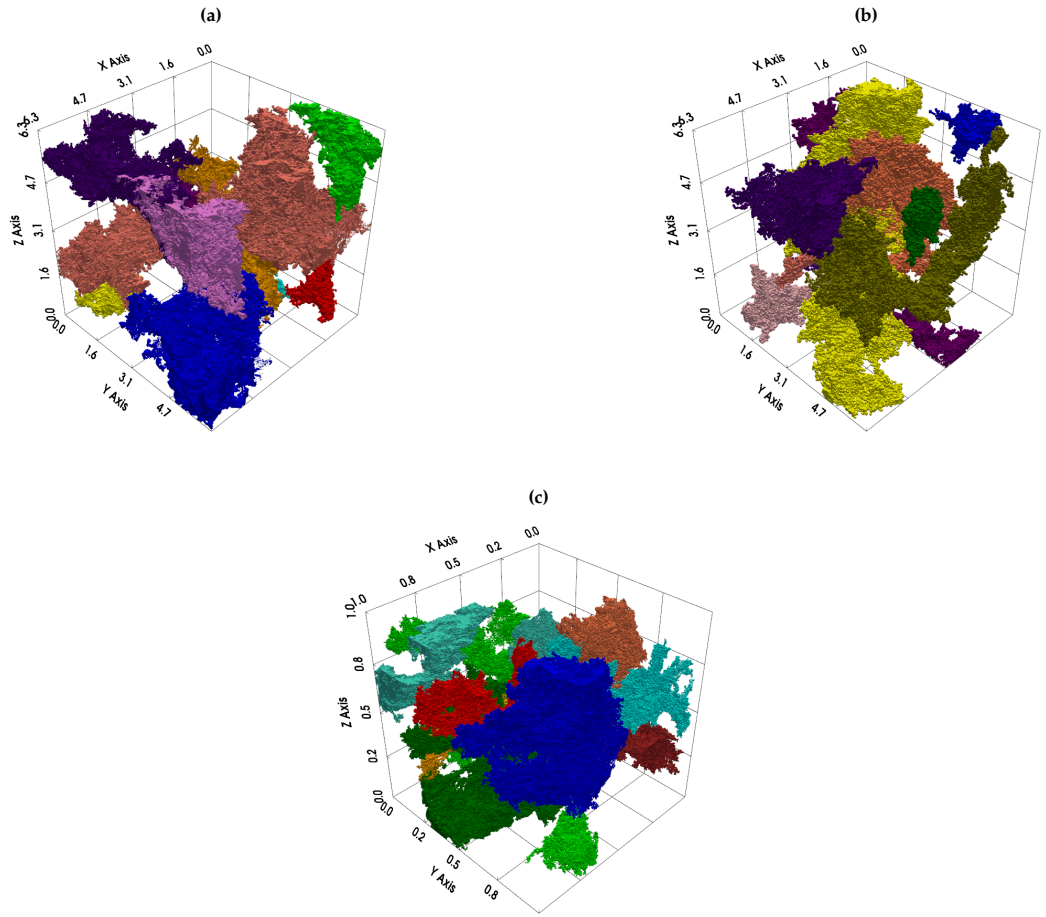


Figure 7.11: Identified structures at different Reynolds numbers (Forced cases). **(a)** $Re_\lambda = 433$. **(b)** $Re_\lambda = 730$. **(c)** $Re_\lambda = 1131$.

Figures 7.10 and 7.11 show the identified structures at different Reynolds numbers. The structures are visualized as volumes within the flow field and are randomly colored to make them easier to distinguish. It can be observed that the identified structures display a variety of shapes and sizes, and they occupy a significant portion of the flow field. At low Reynolds numbers, the structures appear to be more elongated and worm-like, consistent with the findings of Elsinga and Marusic [10]. As the Reynolds number increases, the structures become more blob-like and less elongated. It is also worth noting that the flow for $Re_\lambda < 100$ does not exhibit full scale separation, thus separation between the large scales and the viscous range is not yet established. Therefore, the structures at lower Reynolds numbers could be affected by the viscous effects, which could explain the more elongated shapes. For $Re_\lambda > 100$, the flow satisfies a greater degree of scale separation, and fewer structures show an elongated shape. This suggests that the structures are not affected by the viscous effects and are more representative of the large-scale structures in fully developed flows.

7.2.2. Length scales distributions

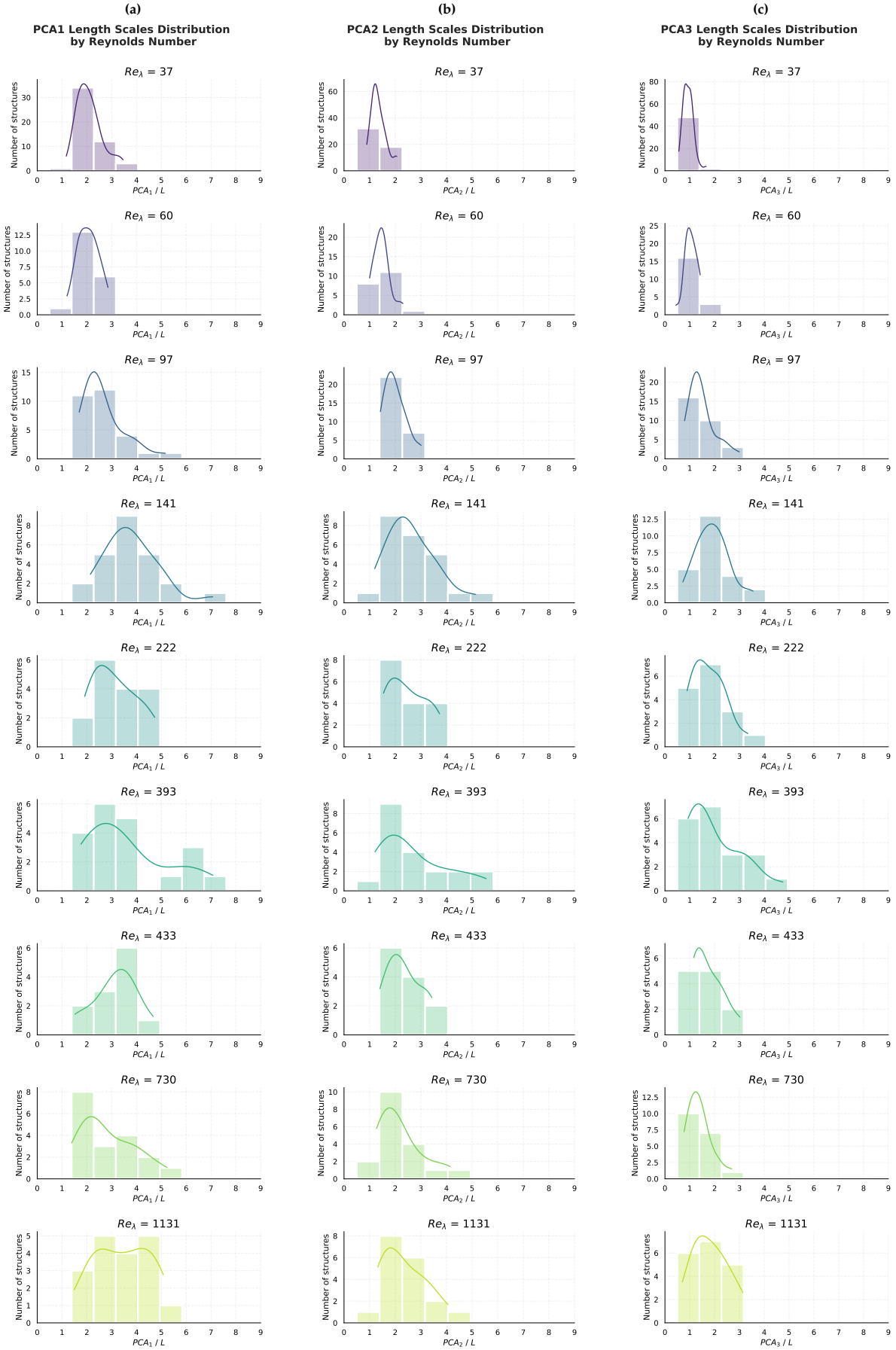


Figure 7.12: PCA length scale distribution of identified structures at different Reynolds numbers.

Figure 7.12 shows the distribution of the principal components of the identified structures at different Reynolds numbers. The length scales are computed using PCA and are normalized by the turbulent integral length scale of the flow field. All structures have all three principal components of the order of the integral length scale of the flow field. This confirms that the identified structures are large-scale features of the flow, and therefore can be classified as large-scale structures. Therefore, it is possible to conclude that the method is able to detect large-scale structures in the flow field, regardless of the Reynolds number. Additionally, the identified structures are of the order of the integral length scale, it confirms that the parameters used in HDBSCAN are universal and do not depend on the Reynolds number. This is an important result, as it suggests that the method can be used to detect large-scale structures in a wide range of datasets without the need for tuning the parameters for each flow. Representative examples of the identified structures at different Reynolds numbers are shown in Appendix A.

7.2.3. Scaling of identified structures

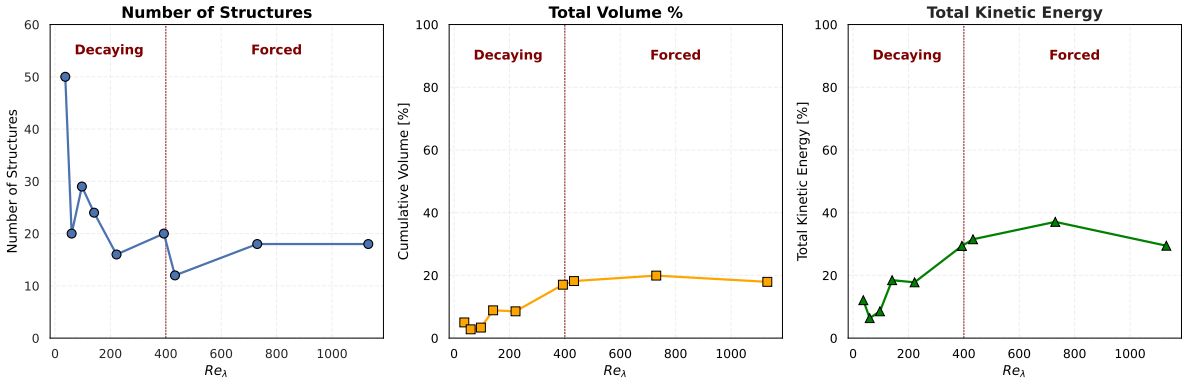


Figure 7.13: Cumulative percentage of volume and kinetic energy contained in large scale structures depending on Re_λ.

Figure 7.13 shows the number of structures detected in each dataset, the cumulative percentage of volume and the kinetic energy contained in the identified structures, as a function of the Reynolds number. The cumulative percentage of volume and kinetic energy is computed by summing the volume and kinetic energy of the identified structures and dividing by the total volume and kinetic energy of the full $2\pi \times 2\pi \times 2\pi$ periodic box. The results show that as the Reynolds number increases, the cumulative percentage of volume and kinetic energy contained in large scale structures also tends to increases for $Re_\lambda \leq 222$. Higher Reynolds numbers show instead an almost constant cumulative percentage of volume and kinetic energy, suggesting that the size and energy content of large scale structures does not change significantly for fully developed turbulent flows. The cumulative kinetic-energy percentage differs by about 5% across datasets, which is reasonable given that each dataset is a snapshot and some variability relative to time averages is expected. The number of identified structures changes significantly across datasets, with no meaningful trend observed, which can also be attributed to the fact that the datasets are snapshots at different integral length scale resolutions and the number of structures can vary significantly over time.

It is also worth observing that between $Re_\lambda = 393$ and $Re_\lambda = 433$ there is no significant change in the cumulative percentage of volume and kinetic energy, despite the fact that the first dataset is decaying, and the second is forced. This suggests that the forcing term does not significantly affect the size and energy content of large scale structures, at least for the Reynolds numbers considered in this study. However, it is important to note that the forcing term can affect the dynamics of the flow and the interaction between structures, which is not considered in this analysis. Therefore, further studies are required to fully understand the effect of the forcing term on large scale structures in turbulent flows.

7.3. Time-resolved dataset analysis

In this section, the analysis of the time-resolved dataset described in Table 6.2 is presented. The analysis focuses on the persistence of the identified structures over time, the variation of kinetic energy, and the temporal evolution of the structures.

An important aspect of this dataset is that the total energy is not constant over time, which has an effect on the Reynolds number Re_λ and on the integral length scale. The time evolution of the total energy, Re_λ, integral length scale, and volume of the identified structures is shown in Figure 7.14. The total energy and Re_λ show a sharp drop over time, which is consistent with the evolution of the integral length scale. The volume of the identified structures shows an opposite trend, showing a significant increase when the total energy drops. This could be attributed to the forcing term, which would increase the injection of energy into the flow to maintain a constant total kinetic energy (E), leading to an increase in the volume of the identified structures.

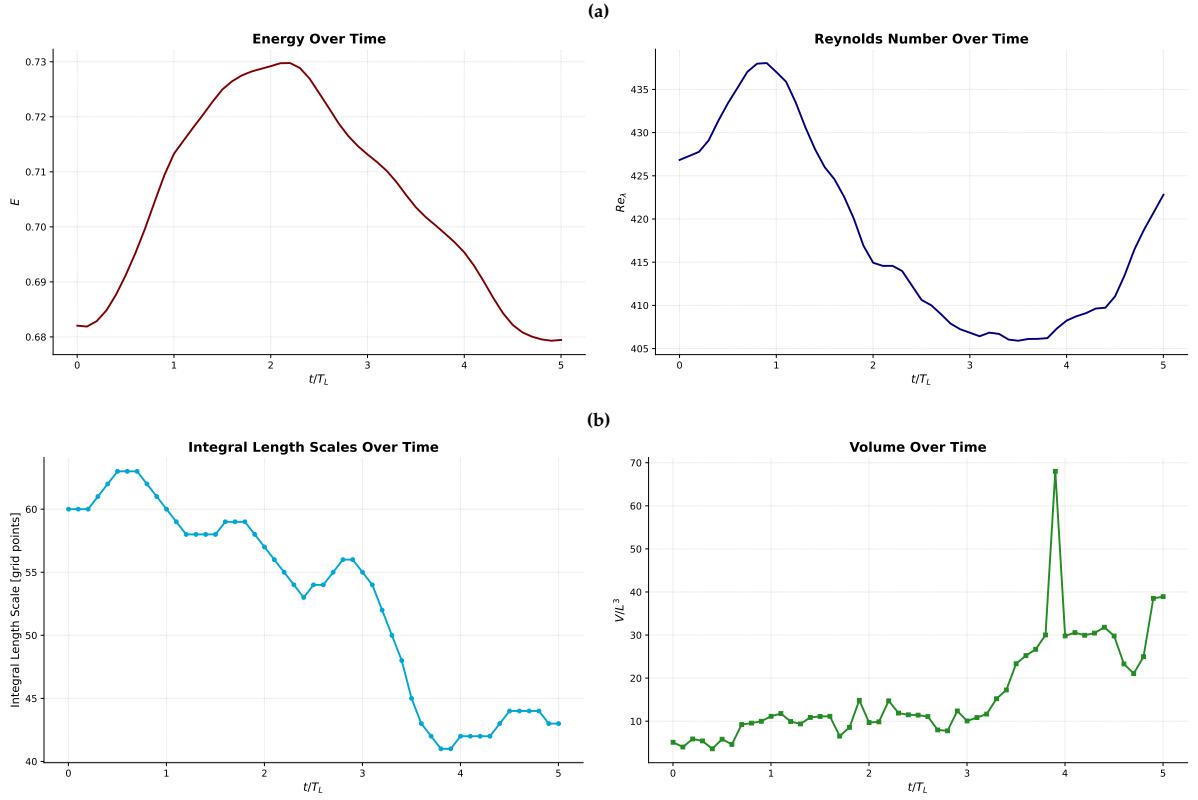


Figure 7.14: Time series analysis of the time-resolved dataset. **(a)** Total energy and Re_λ [79]. **(b)** Computed integral length scale and volume of identified structures.

Since large scale coherent structures are expected to persist over time, the method is applied to a time downsampled time-resolved dataset. The analysis is performed by sampling time snapshot every 0.1 T_L (the large eddy turnover time), which is equivalent to around 1000 time steps of the DNS simulation. In the following analysis, this interval will be referred to as the sampling time. The proposed method is able to detect structures in the time-resolved dataset even with this downsampling. An example of the evolution of a detected structure is shown in Figure 7.15 and its velocity field in Figure 7.16. The detected structure persists for $1.2T_L$, which is consistent with the integral timescale of the flow. The structure is detected at every time sample, and its velocity field is quasi-uniform, confirming that the method is able to detect coherent structures in the time-resolved dataset.

Figure 7.15: Example of identified structures in the time-resolved dataset at $Re_\lambda = 433$. Sampled each 0.2 t/T_L.

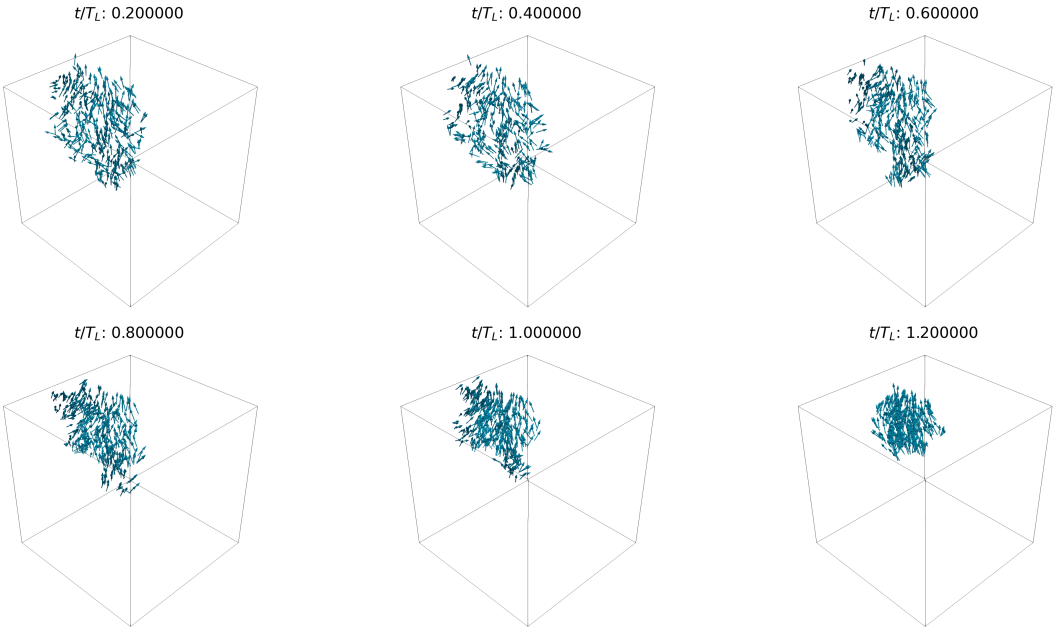


Figure 7.16: Example of velocity field of identified structures in the time-resolved dataset at $Re_\lambda = 433$. Sampled each 0.2 t/T_L.

7.3.1. Persistence of identified structures

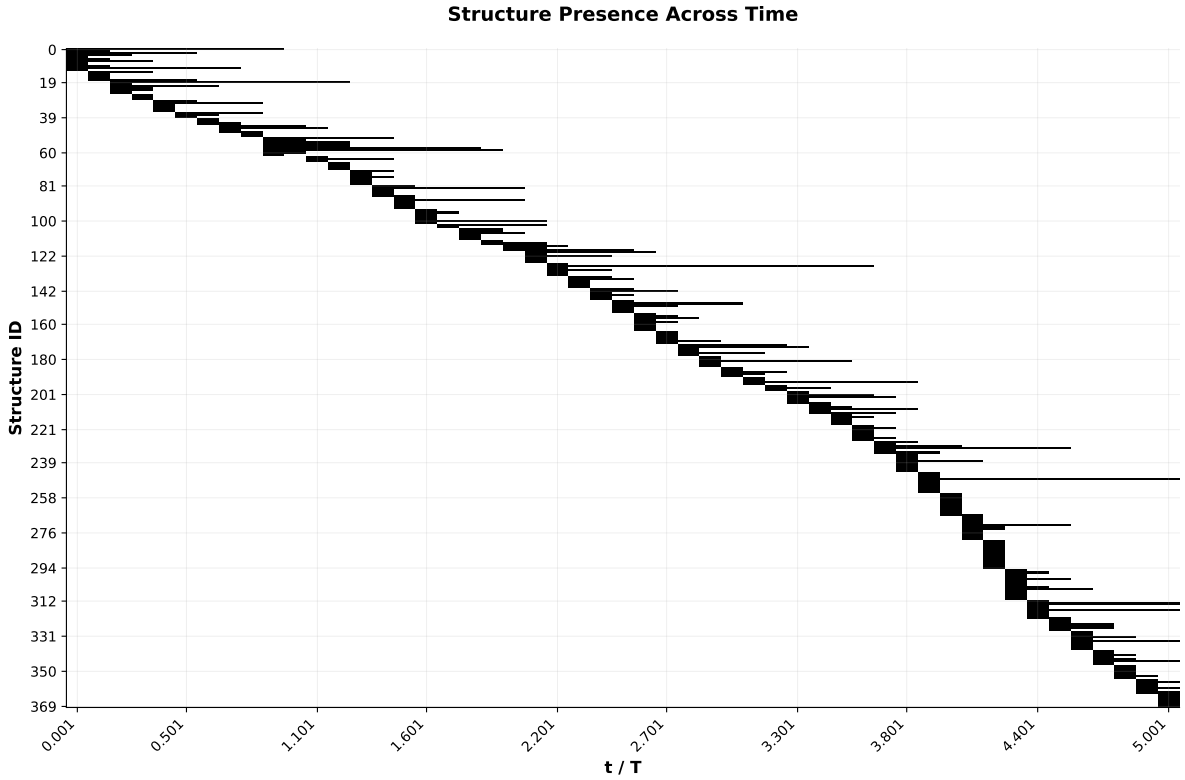


Figure 7.17: Persistence of identified structures in the time-resolved dataset.

Figure 7.17 illustrates the lifespan of the detected coherent structures. The figure's x-axis represents time, and the y-axis identifies each structure. Based on this visualization, only one third of the identified structures persist for more than $0.1T_L$, suggesting that they are predominantly short-lived. This observation is not unexpected, as the chosen time sample of $0.1T_L$ is of the same order as the flow's integral timescale. Additionally, when a structure breaks up, it is considered as two new structures, which can also contribute to the short lifespan of some structures.

Nevertheless, the figure also reveals the presence of some structures with a lifespan exceeding 10 time samples. On average, the mean lifespan of a structure is approximately $0.2T_L$. When structures with a lifespan of a single sampling time are excluded, this average increases to approximately $0.4T_L$. These findings, which characterize coherent structures as long-lived regions, are consistent with the definition of Adrian [19] and provide further validation for the proposed method.

7.3.2. Kinetic energy and volume variation in time

Figure 7.18 illustrates the time evolution of the kinetic energy and volume for structures with a lifespan exceeding four time samples. The y-axis represents the percentage change in kinetic energy and volume between each structure's first and last detected appearance. A significant portion of the structures demonstrates a substantial growth in both kinetic energy and volume over time, with some structures exhibiting an increase in kinetic energy of over 100%.

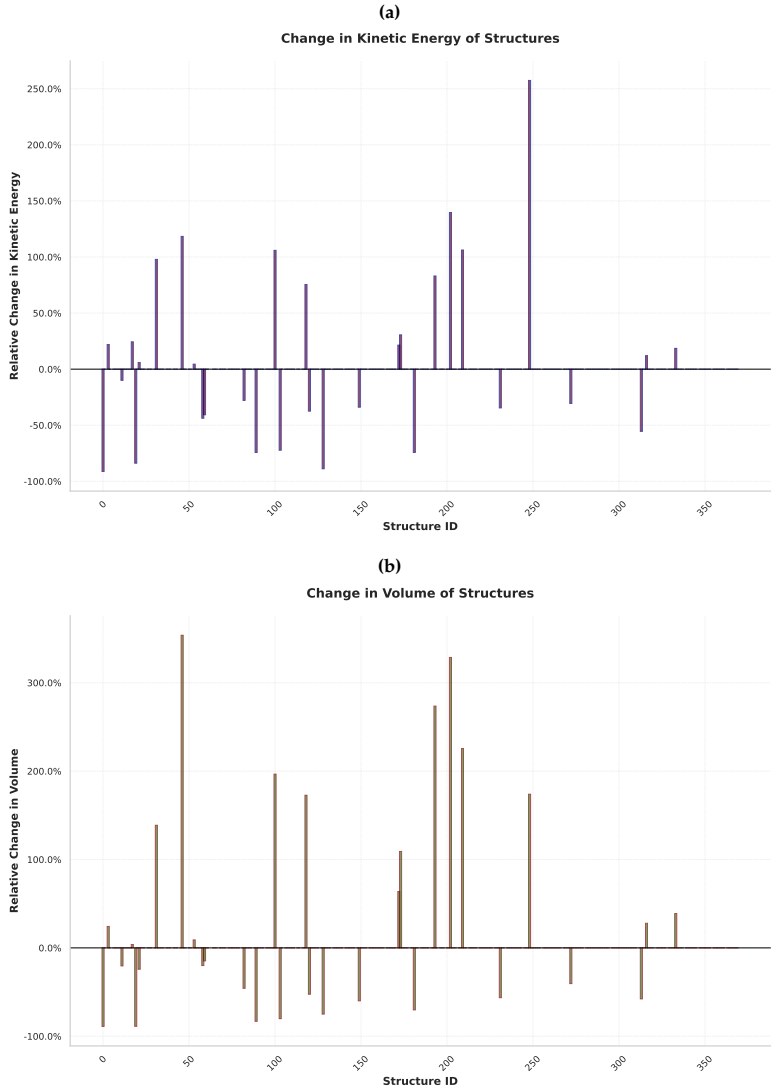


Figure 7.18: Variation of identified structures in the time-resolved dataset. **(a)** Kinetic energy variation. **(b)** Volume variation.

Large-scale structures are generally assumed to decay and break down over time, reducing their kinetic energy. However, this result suggests that some structures within this dataset can grow in both size and kinetic energy over time. This growth behavior is typically associated with 2D turbulent flows [82], where large-scale structures grow due to an inverse energy cascade. While this phenomenon is less common in 3D turbulence, these findings indicate that a similar process may be at play. Specifically, the observed growth could be a result of structures merging with one another. This merging process would lead to a larger and more energetic structure that would be detected as a single entity with an increased volume and kinetic energy over time. This hypothesis is supported by the fact that the flow is constantly being forced, which could provide the necessary energy for the structures to grow and merge instead of simply breaking decaying and breaking up. These findings will be further explored in the next section, where the temporal evolution of the structures is analyzed in more detail.

7.3.3. Temporal evolution of the identified structures

Merging of coherent structures

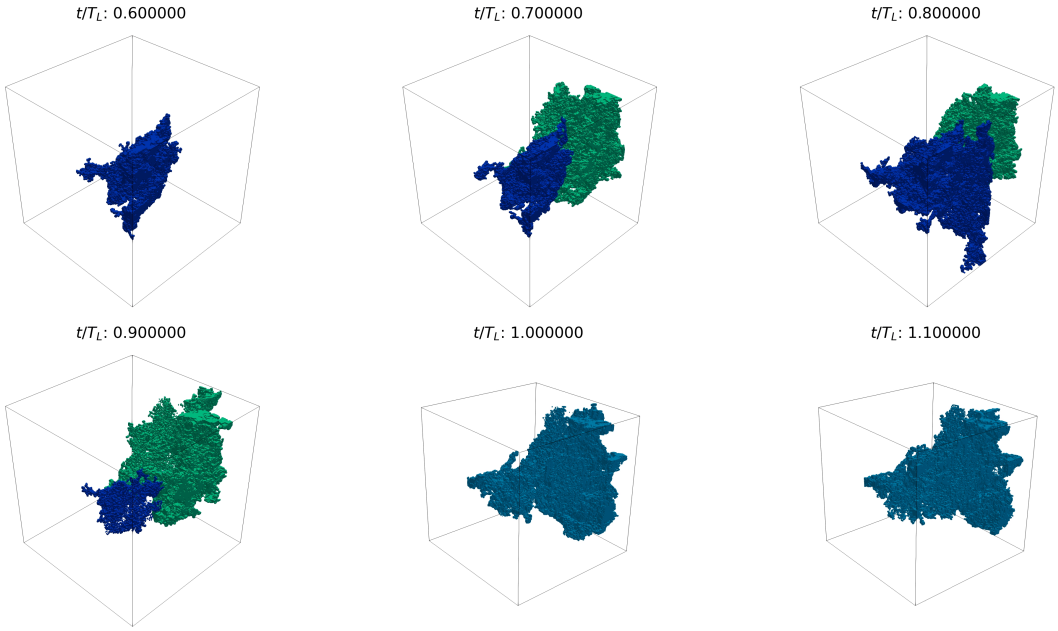


Figure 7.19: Example of two (blue and green) structures merging into a single structure (cyan) in the time-resolved dataset at $Re_\lambda = 433$.

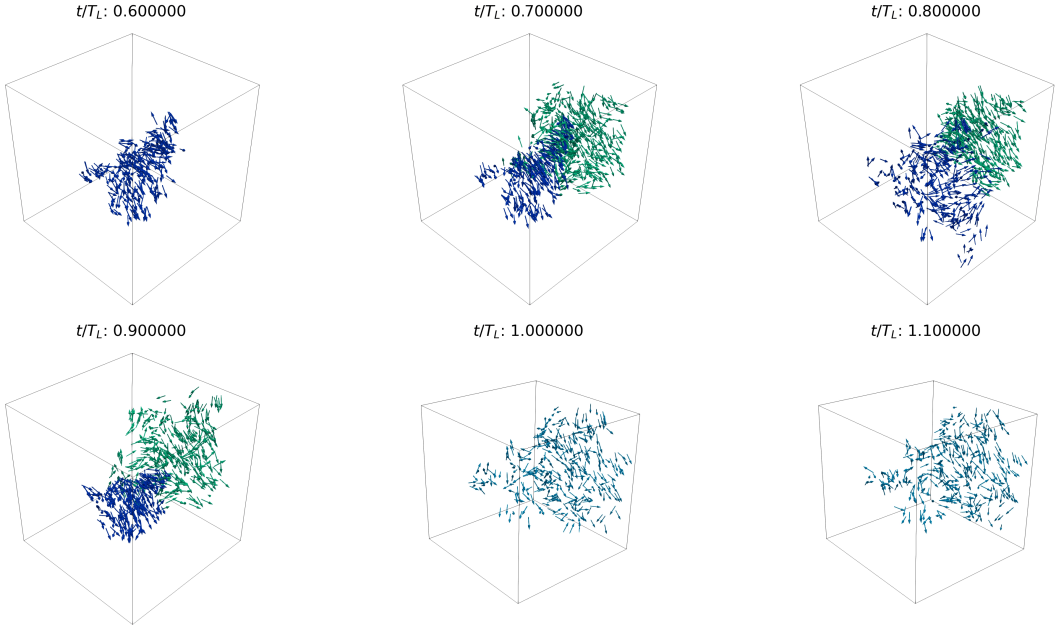


Figure 7.20: Velocity field of two (blue and green) structures merging into a single structure (cyan) in the time-resolved dataset at $Re_\lambda = 433$.

Figure 7.19 illustrates a specific instance of two structures merging in the time-resolved dataset. The evolution of their kinetic energy is reported in Figure 7.21 This observation provides a potential mechanism for the increase in energy and volume discussed previously. Initially, two distinct structures, highlighted in blue and green, are detected at different time steps before they merge into a single, cyan structure. The velocity field of the two initial structures, shown in Figure 7.20, is highly coherent.

However, a significant loss of coherence can be observed in the velocity field of the merged structure, suggesting that the merging process can lead to a decrease in organization.

Furthermore, the merged structure exhibits a short lifespan, which indicates that the merging process induces an instability, resulting in rapid decay or a loss of coherency in every instance. This phenomenon suggests that while merging events can cause a temporary increase in a structure's size and energy, they also introduce an instability that limits its lifespan.

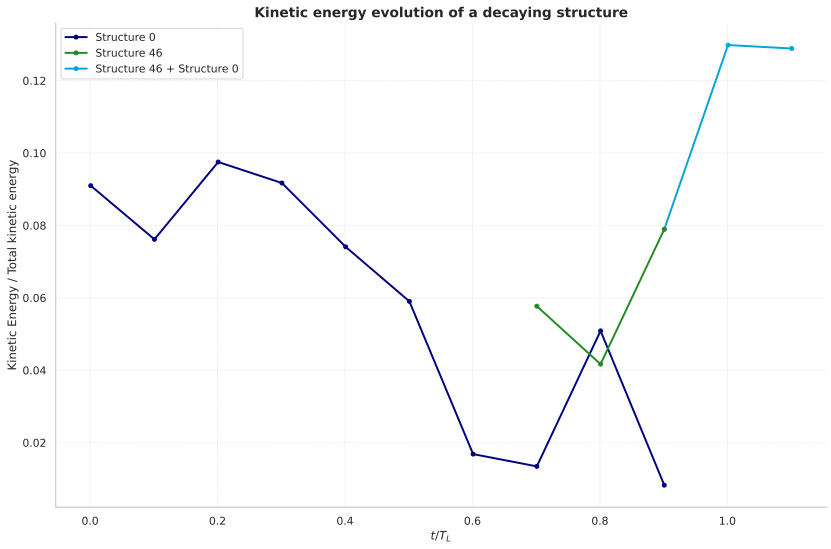


Figure 7.21: Kinetic energy of the two structures before and after merging in the time-resolved dataset at $Re_\lambda = 433$. The blue and green lines represent the kinetic energy of the two initial structures, while the cyan line represents the kinetic energy of the merged structure.

However, a comprehensive interpretation of these findings requires an examination of the limitations of this analysis. The current analysis is based on a time-resolved dataset which has been downsampled temporally and spatially. As previously observed in Section 5.6.1, this downsamplig can lead to some structure to be merged together due to the reduction in the gaps between them. Therefore, the merging events observed in this analysis could be an artifact of the downsampling process, rather than a true representation of the flow dynamics. Future work should focus on analyzing the time-resolved dataset with a finer spatial resolution to better capture the dynamics of the merging events and to confirm whether they are indeed a result of the downsampling process or a true representation of the flow dynamics. Additionally, the results of the current method should be compared with particle tracking methods to validate the merging events and to better understand the dynamics of the structures in the flow.

Decay of coherent structures

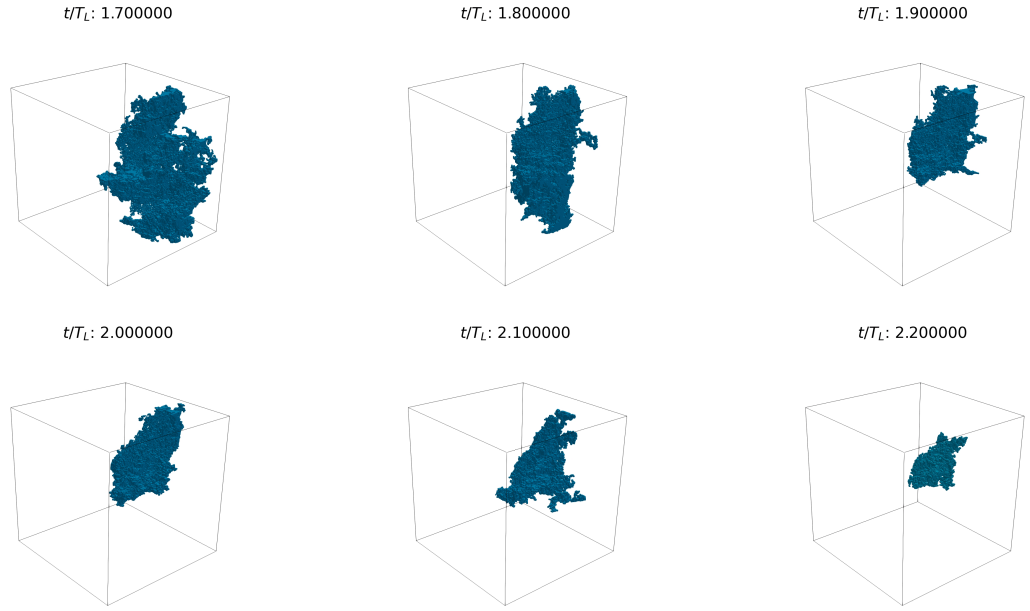


Figure 7.22: Example of decaying structure in the time-resolved dataset at $Re_\lambda = 433$.

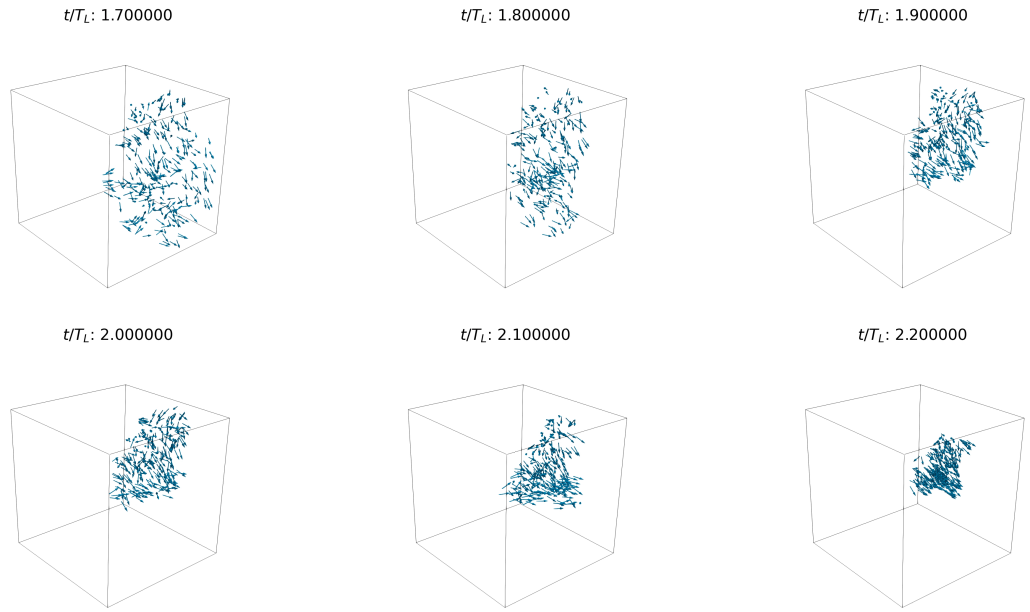


Figure 7.23: Velocity field of the decaying structure in the time-resolved dataset at $Re_\lambda = 433$.

Figure 7.22 shows an example of a decaying structure in the time-resolved dataset. The structure is detected at different time steps. Its velocity field is shown in Figure 7.23 and its kinetic energy in Figure 7.24. The structure shows a significant decrease in size over time, which is consistent with the expected behavior of large-scale structures in the cascade process [82]. The last time the sample showed a structure with a volume of around $0.43L^3$, which is close to the minimum volume threshold used in the detection algorithm. Therefore, it is possible that the structure continues the decay process even after it is no longer detected.

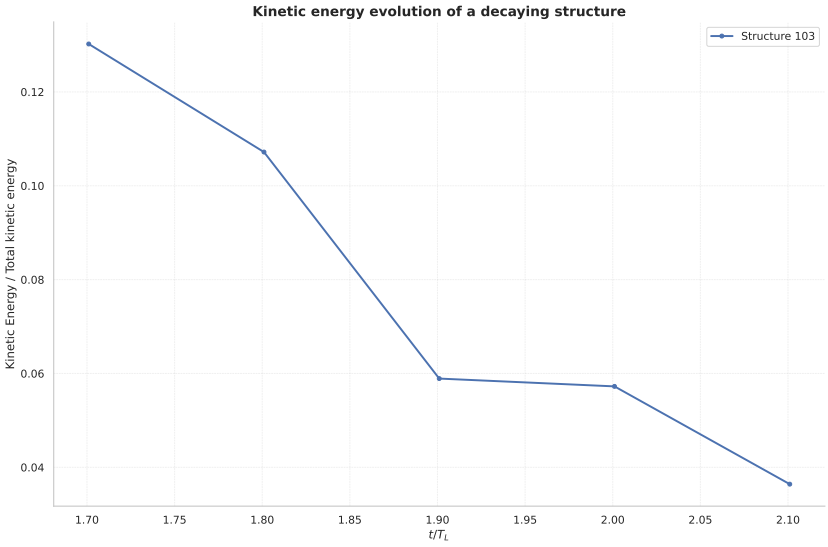
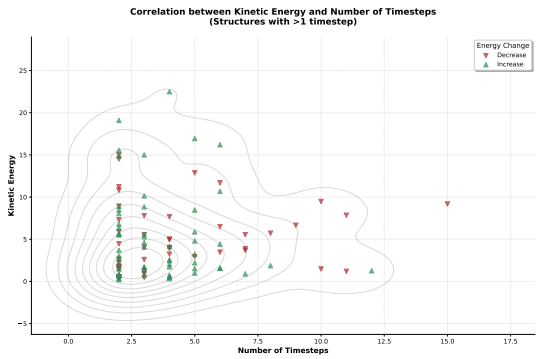


Figure 7.24: Kinetic energy of the decaying structure in the time-resolved dataset at $Re_\lambda = 433$.

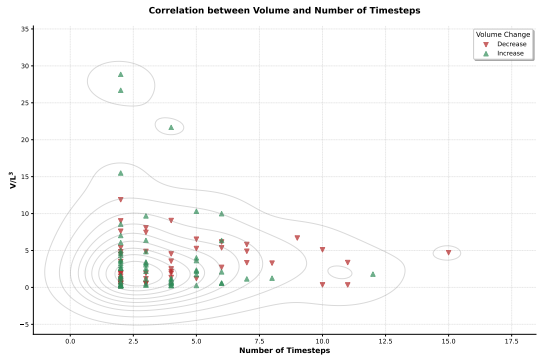
The decay of structures in turbulence is traditionally studied using Fourier analysis, a powerful tool for investigating energy transfer. However, this method has a key limitation: It can only analyze the energy transfer between predefined Fourier modes. In contrast, the proposed method provides a more comprehensive approach by detecting structures directly from the flow field without relying on these predefined modes.

This allows for a more detailed analysis of the decay process, as it directly captures the dynamics of the structures. The results show that decay primarily occurs at the structures' edges. While a detailed analysis of the decay mechanism is beyond the scope of this thesis, two potential explanations are possible. The decay could be the result of either a loss of kinetic energy or a loss of coherency in their velocity field at the edges of the large-scale structures. Both of these theories are consistent with an increase in the wave number used to describe the structure in Fourier space. Future work should therefore focus on a more detailed analysis of the decay process, including its underlying mechanics and the interactions between structures during decay.

7.3.4. Correlation between initial volume, kinetic energy and time persistence



(a) Correlation between initial kinetic energy and time persistence of identified structures.



(b) Correlation between initial volume and time persistence of identified structures.

Figure 7.25: Correlation between initial quantities and time persistence of identified structures.

Figure 7.25 displays the correlation between the initial kinetic energy or volume of the identified structures and their lifespan. The y-axis shows the initial kinetic energy or volume, while the x-axis represents the structure's lifespan. The results indicate that most structures have a lifespan between 0.2

and 0.3 T_L.

An interesting observation is that structures with longer lifespans tend to be decaying over time and possess more contained initial kinetic energy and volume. Conversely, structures that contain a significant amount of kinetic energy and volume at the first time sample tend to have a shorter lifespan. This suggests that very large and energetic structures, such as those that might be generated by the forcing term or merging of structures, are more unstable and have a shorter overall lifespan.

8

Conclusion

The study of turbulence, a phenomenon ubiquitous in nature and engineering, has long been challenged due to its chaotic and multi-scale nature. While classical spectral approaches provide a statistical framework, they often overlook the physical mechanism in real space that governs the energy transfer, stemming from the behavior of large-scale coherent structures. This thesis was motivated by the need for a systematic, consistent, and reproducible method to identify and analyze these large-scale coherent structures in homogeneous isotropic turbulence (HIT). The primary research objective was to develop such a methodology and use it to answer fundamental questions about the nature, behavior, and impact of these structures.

8.1. Conclusions

This section summarizes the key findings of the report, structuring the discussion around the research questions that guided the investigation.

How can large-scale structures in homogeneous isotropic turbulence be identified in a consistent and reproducible way?

This thesis has successfully developed and validated a novel, automated methodology for identifying large-scale structures in HIT flows. The method is built on the HDBSCAN algorithm, which groups points in the flow field based on the similarity of their velocity vectors. The core of the approach is to define a feature space using the three Cartesian velocity components, which circumvents the periodicity and distortion issues associated with spherical coordinate systems used in previous methods [10, 36].

The key features of this method are:

1. **Consistency**: Unlike methods that rely on user-defined thresholds or the manual selection of histogram peaks, this approach is data-driven. The primary parameters, `min_cluster_size` and `min_samples`, have a clear and predictable influence on the coherency of the detected structures, allowing them to be set systematically based on domain size rather than a posteriori.
2. **No previous assumptions**: By clustering in velocity space and subsequently enforcing spatial continuity in a post-processing step, the method does not make any assumptions regarding the shape of the structures.
3. **Robustness**: The method has proven to be robust against variations in downsampling and key hyperparameters. Validation against the manual extractions of Elsinga and Marusic [10] showed a strong qualitative agreement in the identification of the most significant structures.

In summary, the developed HDBSCAN based framework provides a consistent, reproducible, and automated tool to successfully identify large-scale coherent structures.

What are the characteristics of large-scale structures in homogeneous isotropic turbulence?

Applying the methodology to a high-Reynolds-number dataset ($Re_\lambda = 1131$) revealed several defining characteristics of the identified structures.

The structures are predominantly anisotropic and irregularly shaped, as the majority of structures exhibited elongation in at least one direction. Furthermore, these structures are highly energetic, as they consistently contain a larger fraction of the flow's total kinetic energy compared to the volume they occupy. The velocity field within each structure was confirmed to be quasi-uniform, consistent with the definition of coherent structures by Hussain [18].

How do these characteristics of these structures change with the Reynolds number?

The analysis was performed across a wide range of Reynolds numbers ($Re_\lambda \approx 37 - 1131$). The most significant finding from this analysis is the scaling behavior. For fully scale separated turbulent flows ($Re_\lambda \geq 222$), the cumulative volume and kinetic energy contained within the identified structures become nearly constant. This suggests that the large-scale organization of the flow reaches a form of statistical equilibrium, where the overall contribution of these structures to the flow's volume and energy content becomes independent of the Reynolds number.

Is there any observable phenomena that can be attributed to the presence of large-scale structures?

This research provides strong evidence that the interaction between adjacent large-scale structures is a primary mechanism for generating regions of intense energy dissipation. Visualizations clearly show that sheet-like regions of high dissipation ($< \epsilon > +5\sigma_\epsilon$) form in the gaps between neighboring structures.

The velocity fields within these interacting structures are often counter-directed, creating intense shear layers in the fluid between them. It is within these shear layers that the dissipation rate peaks significantly, confirming theoretical expectations and previous studies [10, 9]. This observation provides a direct physical link between large-scale motions and the small-scale intermittent events that are crucial to understanding turbulent energy transfer.

What are the observable dynamics of large-scale structures?

The analysis of a time-resolved dataset provided new insight into the life cycle of these structures. Key observation include:

- **Persistence**: While many structures are transient, a significant portion persists for durations on the order of the large-eddy turnover time (T_L), with a mean lifespan of approximately $0.4T_L$ for structures lasting more than one time step. This confirms their temporal coherence, a defining trait of coherent structures as highlighted by Adrian [19].
- **Decay**: In line with classical cascade theory, many structures were observed to decay over time, gradually shrinking in volume. The decay process appears to happen primarily at the structures edges.
- **Merging**: An unexpected finding was the observation of structures growing significantly in both volume and kinetic energy. This growth was found to be a result of the merging of two or more distinct structures into a single, larger entity. However, this process appears to induce instability; the newly formed, larger structures consistently exhibit a loss of internal velocity coherence and have a very short lifespan.
- **Lifespan Observations**: Very large and energetic structures, possibly formed by merging or direct forcing, tend to be unstable and have shorter lifespans.

8.2. Future work

This thesis developed a robust framework for the identification of large-scale structures in turbulence. By applying this method, an initial characterization of the geometry, scaling, and dynamics of these structures was given. This provided new insights into the physical mechanisms that govern the energy cascade. The findings confirm the crucial role of the large-scale structures in generating dissipation and have uncovered their complex dynamics.

Future work should focus on confirming and further exploring these findings. The following directions are proposed:

1. This work qualitatively confirmed that regions of high energy dissipation form in the shear layers between interacting large-scale structures. A crucial next step is to quantify this relationship. Future studies could develop metrics to correlate structure proximity, relative velocity, and orientation with the intensity and geometry of dissipation events, providing a more precise physical model of their role in the energy cascade.
2. The analysis of the time-resolved dataset revealed complex dynamics, including structure merging and decay. To validate and generalize these findings, this analysis should be extended to time-resolved datasets at different Reynolds numbers (Re_λ) and forcing schemes. This would clarify whether the observed merging events are a universal feature of HIT flows or specific to the forcing mechanisms of the dataset studied.
3. The current study noted that observed merging events might be partially influenced by the spatial and temporal downsampling of the dataset. Future research should analyze these events using datasets with higher fidelity to distinguish numerical artifacts from physical phenomena.
4. To further validate the temporal evolution of structures identified by the HDBSCAN based method, a comparative study using Lagrangian particle tracking should be conducted. Tracking fluid particles would provide an independent confirmation of the observed merging and decay dynamics.

References

[1] Mordecai-Mark Mac Low and Ralf S. Klessen. "Control of Star Formation by Supersonic Turbulence". Version 2. In: (2003). DOI: 10.48550/ARXIV.ASTRO-PH/0301093.

[2] ZhenHua Wan, Lin Zhou, and DeJun Sun. "A Study on Large Coherent Structures and Noise Emission in a Turbulent Round Jet". In: *Sci. China Phys. Mech. Astron.* 57.8 (Aug. 2014), pp. 1552–1562. DOI: 10.1007/s11433-013-5291-2.

[3] F. T. M. Nieuwstadt et al. *Turbulence: Introduction to Theory and Applications of Turbulent Flows*. SpringerLink Bücher. Cham: Springer, 2016. 284 pp. DOI: 10.1007/978-3-319-31599-7.

[4] "The Local Structure of Turbulence in Incompressible Viscous Fluid for Very Large Reynolds Numbers". In: *Proc. R. Soc. Lond. A* 434.1890 (July 8, 1991), pp. 9–13. DOI: 10.1098/rspa.1991.0075.

[5] Lewis Fry Richardson and Peter Lynch. *Weather Prediction by Numerical Process*. 2nd ed. Cambridge University Press, Aug. 13, 2007. DOI: 10.1017/CBO9780511618291.

[6] J. Andrzej Domaradzki and Robert S. Rogallo. "Local Energy Transfer and Nonlocal Interactions in Homogeneous, Isotropic Turbulence". In: *Physics of Fluids A: Fluid Dynamics* 2.3 (Mar. 1, 1990), pp. 413–426. DOI: 10.1063/1.857736.

[7] Ye Zhou. "Degrees of Locality of Energy Transfer in the Inertial Range". In: *Physics of Fluids A: Fluid Dynamics* 5.5 (May 1, 1993), pp. 1092–1094. DOI: 10.1063/1.858593.

[8] J. Andrzej Domaradzki. "Nonlocal Triad Interactions and the Dissipation Range of Isotropic Turbulence". In: *Physics of Fluids A: Fluid Dynamics* 4.9 (Sept. 1, 1992), pp. 2037–2045. DOI: 10.1063/1.858373.

[9] Gerrit E. Elsinga, Takashi Ishihara, and Julian C. R. Hunt. "Extreme Dissipation and Intermittency in Turbulence at Very High Reynolds Numbers". In: *Proc. R. Soc. A.* 476.2243 (Nov. 2020), p. 20200591. DOI: 10.1098/rspa.2020.0591.

[10] G. E. Elsinga and I. Marusic. "Universal Aspects of Small-Scale Motions in Turbulence". In: *J. Fluid Mech.* 662 (Nov. 10, 2010), pp. 514–539. DOI: 10.1017/S0022112010003381.

[11] Danah Park and Adrián Lozano-Durán. "The Coherent Structure of the Energy Cascade in Isotropic Turbulence". In: *Sci Rep* 15.1 (Jan. 2, 2025), p. 14. DOI: 10.1038/s41598-024-80698-3.

[12] G. E. Elsinga et al. "The Scaling of Straining Motions in Homogeneous Isotropic Turbulence". In: *J. Fluid Mech.* 829 (Oct. 25, 2017), pp. 31–64. DOI: 10.1017/jfm.2017.538.

[13] "The Local Structure of Turbulence in Incompressible Viscous Fluid for Very Large Reynolds Numbers". In: *Proc. R. Soc. Lond. A* 434.1890 (July 8, 1991), pp. 9–13. DOI: 10.1098/rspa.1991.0075.

[14] Takashi Ishihara, Yukio Kaneda, and Julian C. R. Hunt. "Thin Shear Layers in High Reynolds Number Turbulence—DNS Results". In: *Flow Turbulence Combust* 91.4 (Dec. 2013), pp. 895–929. DOI: 10.1007/s10494-013-9499-z.

[15] Takashi Ishihara, Yukio Kaneda, and Julian C. R. Hunt. "Thin Shear Layers in High Reynolds Number Turbulence—DNS Results". In: *Flow Turbulence Combust* 91.4 (Dec. 2013), pp. 895–929. DOI: 10.1007/s10494-013-9499-z.

[16] "Production and Dissipation of Vorticity in a Turbulent Fluid". In: *Proc. R. Soc. Lond. A* 164.916 (Jan. 7, 1938), pp. 15–23. DOI: 10.1098/rspa.1938.0002.

[17] Perry L. Johnson. "On the Role of Vorticity Stretching and Strain Self-Amplification in the Turbulence Energy Cascade". In: *J. Fluid Mech.* 922 (Sept. 10, 2021), A3. DOI: 10.1017/jfm.2021.490.

[18] A. K. M. Fazle Hussain. "Coherent Structures and Turbulence". In: *J. Fluid Mech.* 173 (Dec. 1986), pp. 303–356. DOI: 10.1017/S0022112086001192.

[19] Ronald J. Adrian. "Hairpin Vortex Organization in Wall Turbulence". In: *Physics of Fluids* 19.4 (Apr. 1, 2007), p. 041301. DOI: 10.1063/1.2717527.

[20] S. J. Kline et al. “The Structure of Turbulent Boundary Layers”. In: *J. Fluid Mech.* 30.4 (Dec. 22, 1967), pp. 741–773. DOI: 10.1017/S0022112067001740.

[21] Jinhee Jeong and Fazle Hussain. “On the Identification of a Vortex”. In: *J. Fluid. Mech.* 285 (Feb. 25, 1995), pp. 69–94. DOI: 10.1017/S0022112095000462.

[22] Chaoqun Liu et al. “Third Generation of Vortex Identification Methods: Omega and Liutex/Rortex Based Systems”. In: *J Hydrodyn* 31.2 (Apr. 2019), pp. 205–223. DOI: 10.1007/s42241-019-0022-4.

[23] G Berkooz, P Holmes, and J L Lumley. “The Proper Orthogonal Decomposition in the Analysis of Turbulent Flows”. In: *Annu. Rev. Fluid Mech.* 25.1 (Jan. 1993), pp. 539–575. DOI: 10.1146/annurev.fl.25.010193.002543.

[24] Cheng Chi et al. “Identification and Analysis of Very-Large-Scale Turbulent Motions Using Multiscale Proper Orthogonal Decomposition”. In: *Phys. Rev. Fluids* 7.8 (Aug. 15, 2022), p. 084603. DOI: 10.1103/PhysRevFluids.7.084603.

[25] Rui Yang et al. “Data-driven Identification of the Spatiotemporal Structure of Turbulent Flows by Streaming Dynamic Mode Decomposition”. In: *GAMM-Mitteilungen* 45.1 (Mar. 2022), e202200003. DOI: 10.1002/gamm.202200003.

[26] Tomas W. Muld, Gunilla Efraimsson, and Dan S. Henningson. “Flow Structures around a High-Speed Train Extracted Using Proper Orthogonal Decomposition and Dynamic Mode Decomposition”. In: *Computers & Fluids* 57 (Mar. 2012), pp. 87–97. DOI: 10.1016/j.compfluid.2011.12.012.

[27] Marie Farge et al. “Coherent Vortex Extraction in Three-Dimensional Homogeneous Turbulence: Comparison between CVS-wavelet and POD-Fourier Decompositions”. In: *Physics of Fluids* 15.10 (Oct. 1, 2003), pp. 2886–2896. DOI: 10.1063/1.1599857.

[28] Eric D. Siggia. “Numerical Study of Small-Scale Intermittency in Three-Dimensional Turbulence”. In: *J. Fluid Mech.* 107 (-1 June 1981), p. 375. DOI: 10.1017/S002211208100181X.

[29] F. Moisy and J. Jiménez. “Geometry and Clustering of Intense Structures in Isotropic Turbulence”. In: *J. Fluid Mech.* 513 (Aug. 25, 2004), pp. 111–133. DOI: 10.1017/S0022112004009802.

[30] Stephen B. Pope. *Turbulent Flows*. 1. publ., 12. print. Cambridge: Cambridge Univ. Press, 2015. 771 pp.

[31] Susumu Goto. “A Physical Mechanism of the Energy Cascade in Homogeneous Isotropic Turbulence”. In: *J. Fluid Mech.* 605 (June 25, 2008), pp. 355–366. DOI: 10.1017/S0022112008001511.

[32] T. Leung, N. Swaminathan, and P. A. Davidson. “Geometry and Interaction of Structures in Homogeneous Isotropic Turbulence”. In: *J. Fluid Mech.* 710 (Nov. 10, 2012), pp. 453–481. DOI: 10.1017/jfm.2012.373.

[33] N. A. K. Doan et al. “Scale Locality of the Energy Cascade Using Real Space Quantities”. In: *Phys. Rev. Fluids* 3.8 (Aug. 6, 2018), p. 084601. DOI: 10.1103/PhysRevFluids.3.084601.

[34] Mitsuaki Horiguchi et al. “Large-Scale Turbulence Structures and Their Contributions to the Momentum Flux and Turbulence in the Near-Neutral Atmospheric Boundary Layer Observed from a 213-m Tall Meteorological Tower”. In: *Boundary-Layer Meteorol* 144.2 (Aug. 2012), pp. 179–198. DOI: 10.1007/s10546-012-9718-5.

[35] Iván Bermejo-Moreno and D. I. Pullin. “On the Non-Local Geometry of Turbulence”. In: *J. Fluid Mech.* 603 (May 25, 2008), pp. 101–135. DOI: 10.1017/S002211200800092X.

[36] A.K.M. Ramanna. “Identification of Large-Scale Structures in Turbulence”. Delft University of Technology, 2023.

[37] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. Version 1. 2015. DOI: 10.48550/ARXIV.1505.04597. URL: https://arxiv.org/abs/1505.04597 (visited on 05/15/2025). Pre-published.

[38] Vladimir Estivill-Castro. “Why so Many Clustering Algorithms: A Position Paper”. In: *SIGKDD Explor. Newsl.* 4.1 (June 2002), pp. 65–75. DOI: 10.1145/568574.568575.

[39] Jiawei Han, Micheline Kamber, and Jian Pei. *Data Mining: Concepts and Techniques*. 3rd ed. Morgan Kaufmann Series in Data Management Systems. Amsterdam Boston: Elsevier/Morgan Kaufmann, 2012.

[40] Peter J. Rousseeuw. "Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis". In: *Journal of Computational and Applied Mathematics* 20 (Nov. 1987), pp. 53–65. DOI: 10.1016/0377-0427(87)90125-7.

[41] Zhen Hu and Raj Bhatnagar. "Clustering Algorithm Based on Mutual K-nearest Neighbor Relationships". In: *Statistical Analysis and Data Mining: An ASA Data Science Journal* 5.2 (2012), pp. 100–113. DOI: 10.1002/sam.10149.

[42] Zeming Wei et al. "An Improved Method for Coherent Structure Identification Based on Mutual K-nearest Neighbors". In: *Journal of Turbulence* 23.11–12 (Dec. 2, 2022), pp. 655–673. DOI: 10.1080/14685248.2022.2159421.

[43] Martin Ester et al. "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise". In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. KDD'96. Portland, Oregon: AAAI Press, 1996, pp. 226–231.

[44] Erich Schubert et al. "DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN". In: *ACM Trans. Database Syst.* 42.3 (Sept. 30, 2017), pp. 1–21. DOI: 10.1145/3068335.

[45] Ricardo J. G. B. Campello, Davoud Moulavi, and Joerg Sander. "Density-Based Clustering Based on Hierarchical Density Estimates". In: *Advances in Knowledge Discovery and Data Mining*. Ed. by Jian Pei et al. Red. by David Hutchison et al. Vol. 7819. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 160–172. DOI: 10.1007/978-3-642-37456-2_14.

[46] Ricardo J. G. B. Campello et al. "Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection". In: *ACM Trans. Knowl. Discov. Data* 10.1 (July 27, 2015), pp. 1–51. DOI: 10.1145/2733381.

[47] Claudia Malzer and Marcus Baum. "A Hybrid Approach To Hierarchical Density-based Cluster Selection". In: *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*. Sept. 14, 2020, pp. 223–228. DOI: 10.1109/MFI49285.2020.9235263. arXiv: 1911.02282 [cs].

[48] *The Hdbscan Clustering Library — Hdbscan 0.8.1 Documentation*. URL: https://hdbscan.readthedocs.io/en/latest/index.html (visited on 05/18/2025).

[49] D. W. Muller and G. Sawitzki. "Excess Mass Estimates and Tests for Multimodality". In: *Journal of the American Statistical Association* 86.415 (Sept. 1991), p. 738. DOI: 10.2307/2290406. JSTOR: 2290406.

[50] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. New York: Springer, 2006. 738 pp.

[51] Hugo Touvron et al. *Training Data-Efficient Image Transformers & Distillation through Attention*. Jan. 15, 2021. DOI: 10.48550/arXiv.2012.12877. arXiv: 2012.12877 [cs]. URL: http://arxiv.org/abs/2012.12877 (visited on 05/21/2025). Pre-published.

[52] Zongwei Zhou et al. *UNet++: A Nested U-Net Architecture for Medical Image Segmentation*. July 18, 2018. DOI: 10.48550/arXiv.1807.10165. arXiv: 1807.10165 [cs]. URL: http://arxiv.org/abs/1807.10165 (visited on 05/20/2025). Pre-published.

[53] Razvan-Gabriel Dumitru, Darius Peteleaza, and Catalin Craciun. "Using DUCK-Net for Polyp Image Segmentation". In: *Sci Rep* 13.1 (June 16, 2023), p. 9803. DOI: 10.1038/s41598-023-36940-5.

[54] Özgün Çiçek et al. *3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation*. Version 1. 2016. DOI: 10.48550/ARXIV.1606.06650. URL: https://arxiv.org/abs/1606.06650 (visited on 05/20/2025). Pre-published.

[55] Kaiming He et al. *Mask R-CNN*. Jan. 24, 2018. DOI: 10.48550/arXiv.1703.06870. arXiv: 1703.06870 [cs]. URL: http://arxiv.org/abs/1703.06870 (visited on 05/20/2025). Pre-published.

[56] Tsung-Yi Lin et al. *Microsoft COCO: Common Objects in Context*. Version 3. 2014. DOI: 10.48550/ARXIV.1405.0312. URL: https://arxiv.org/abs/1405.0312 (visited on 05/20/2025). Pre-published.

[57] Joseph Redmon et al. *You Only Look Once: Unified, Real-Time Object Detection*. May 9, 2016. DOI: 10.48550/arXiv.1506.02640. arXiv: 1506.02640 [cs]. URL: http://arxiv.org/abs/1506.02640 (visited on 05/20/2025). Pre-published.

[58] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. *YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors*. Version 1. 2022. DOI: 10.48550/ARXIV.2207.02696. URL: https://arxiv.org/abs/2207.02696 (visited on 05/20/2025). Pre-published.

[59] Fenqiang Zhao et al. "Spherical U-Net on Cortical Surfaces: Methods and Applications". In: *Information Processing in Medical Imaging*. Ed. by Albert C. S. Chung et al. Vol. 11492. Cham: Springer International Publishing, 2019, pp. 855–866. DOI: 10.1007/978-3-030-20351-1_67.

[60] Qing-Yang Shen et al. "Training Real-Time Panoramic Object Detectors with Virtual Dataset". In: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Toronto, ON, Canada: IEEE, June 6, 2021, pp. 1520–1524. DOI: 10.1109/ICASSP39728.2021.9414503.

[61] Ibrahim Djemai et al. *Extending 2D Saliency Models for Head Movement Prediction in 360-Degree Images Using CNN-based Fusion*. Feb. 21, 2020. DOI: 10.48550/arXiv.2002.09196. arXiv: 2002.09196 [eess]. URL: http://arxiv.org/abs/2002.09196 (visited on 07/11/2025). Pre-published.

[62] Mark R. Calabretta and Boudewijn F. Roukema. "Mapping on the HEALPix Grid". In: *Monthly Notices of the Royal Astronomical Society* 381.2 (Oct. 2007), pp. 865–872. DOI: 10.1111/j.1365-2966.2007.12297.x.

[63] Rolf Westerteiger, Andreas Gerndt, and Bernd Hamann. "Spherical Terrain Rendering Using the Hierarchical HEALPix Grid". In: *OASIcs, Volume 27, VLUDS 2011* 27 (2013). Ed. by Christoph Garth, Ariane Middel, and Hans Hagen, pp. 13–23. DOI: 10.4230/OASICS.VLUDS.2011.13.

[64] Fenqiang Zhao et al. "Spherical U-Net on Cortical Surfaces: Methods and Applications". In: *Information Processing in Medical Imaging*. Ed. by Albert C. S. Chung et al. Vol. 11492. Cham: Springer International Publishing, 2019, pp. 855–866. DOI: 10.1007/978-3-030-20351-1_67.

[65] Alexey Dosovitskiy et al. *An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale*. June 3, 2021. DOI: 10.48550/arXiv.2010.11929. arXiv: 2010.11929 [cs]. URL: http://arxiv.org/abs/2010.11929 (visited on 05/21/2025). Pre-published.

[66] Ashish Vaswani et al. *Attention Is All You Need*. Aug. 2, 2023. DOI: 10.48550/arXiv.1706.03762. arXiv: 1706.03762 [cs]. URL: http://arxiv.org/abs/1706.03762 (visited on 05/21/2025). Pre-published.

[67] José Maurício, Inês Domingues, and Jorge Bernardino. "Comparing Vision Transformers and Convolutional Neural Networks for Image Classification: A Literature Review". In: *Applied Sciences* 13.9 (Apr. 28, 2023), p. 5521. DOI: 10.3390/app13095521.

[68] Alexander Kirillov et al. "Segment Anything". In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023 IEEE/CVF International Conference on Computer Vision (ICCV). Paris, France: IEEE, Oct. 1, 2023, pp. 3992–4003. DOI: 10.1109/ICCV51070.2023.00371.

[69] Nikhila Ravi et al. "SAM 2: Segment Anything in Images and Videos". In: ().

[70] *Maximum_filter — SciPy v1.15.3 Manual*. URL: https://docs.scipy.org/doc/scipy-1.15.3/reference/generated/scipy.ndimage.maximum_filter.html (visited on 05/21/2025).

[71] Charles R. Harris et al. "Array Programming with NumPy". In: *Nature* 585.7825 (Sept. 17, 2020), pp. 357–362. DOI: 10.1038/s41586-020-2649-2.

[72] F. Pedregosa et al. "Scikit-Learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[73] Leland McInnes, John Healy, and Steve Astels. "Hdbscan: Hierarchical Density Based Clustering". In: *JOSS* 2.11 (Mar. 21, 2017), p. 205. DOI: 10.21105/joss.00205.

[74] RAPIDS Development Team. *RAPIDS: Libraries for End to End GPU Data Science*. manual. 2023.

[75] Will Schroeder, Ken Martin, and Bill Lorensen. *The Visualization Toolkit (4th Ed.)* Kitware, 2006.

[76] Thomas H. Cormen, ed. *Introduction to Algorithms*. 3rd ed. Cambridge, Mass: MIT Press, 2009. 1292 pp.

[77] Karl Pearson. "LIII. *On Lines and Planes of Closest Fit to Systems of Points in Space*". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11 (Nov. 1901), pp. 559–572. DOI: 10.1080/14786440109462720.

[78] M. Tanahashi, T. Miyauchi, and J. Ikeda. "Identification of Coherent Fine Scale Structure in Turbulence". In: *IUTAM Symposium on Simulation and Identification of Organized Structures in Flows*. Ed. by J. N. Sørensen, E. J. Hopfinger, and N. Aubry. Red. by R. Moreau. Vol. 52. Dordrecht: Springer Netherlands, 1999, pp. 131–140. DOI: 10.1007/978-94-011-4601-2_12.

[79] Wan Minping et al. *Forced Isotropic Turbulence Data Set (Extended)*. Johns Hopkins Turbulence Databases, 2012. DOI: 10.7281/T1KK98XB.

[80] Takashi Ishihara, Yukio Kaneda, and Julian C. R. Hunt. "Thin Shear Layers in High Reynolds Number Turbulence—DNS Results". In: *Flow Turbulence Combust* 91.4 (Dec. 2013), pp. 895–929. DOI: 10.1007/s10494-013-9499-z.

[81] G. E. Elsinga et al. "The Scaling of Straining Motions in Homogeneous Isotropic Turbulence". In: *J. Fluid Mech.* 829 (Oct. 25, 2017), pp. 31–64. DOI: 10.1017/jfm.2017.538.

[82] Peter Davidson. *Turbulence: An Introduction for Scientists and Engineers*. Oxford University Press, June 1, 2015. DOI: 10.1093/acprof:oso/9780198722588.001.0001.

A

Examples of identified structures for different Reynolds numbers

This appendix presents additional representative examples of detected large-scale structures for different Reynolds numbers. Each example includes the spherical-angle probability density function of the structure, a visualization of the structure within the computational domain, and a vector visualization of the velocity field inside the structure.

A.1. Reynolds number $\mathbf{Re}_\lambda \approx 37$

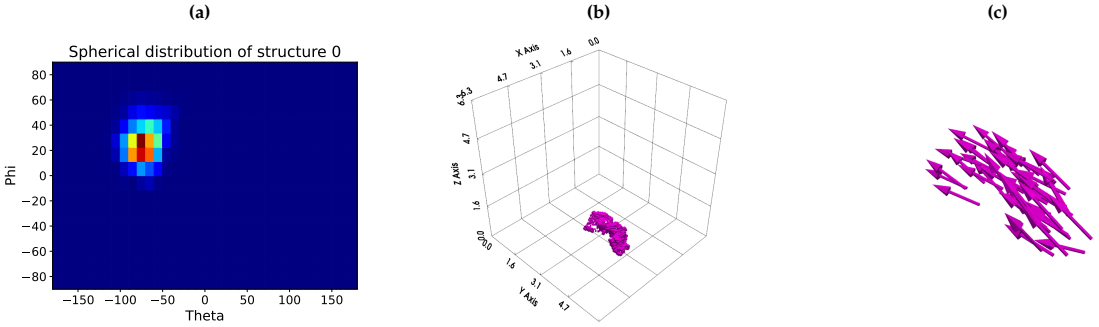


Figure A.1: Example of detected structure in the $Re_\lambda = 37$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

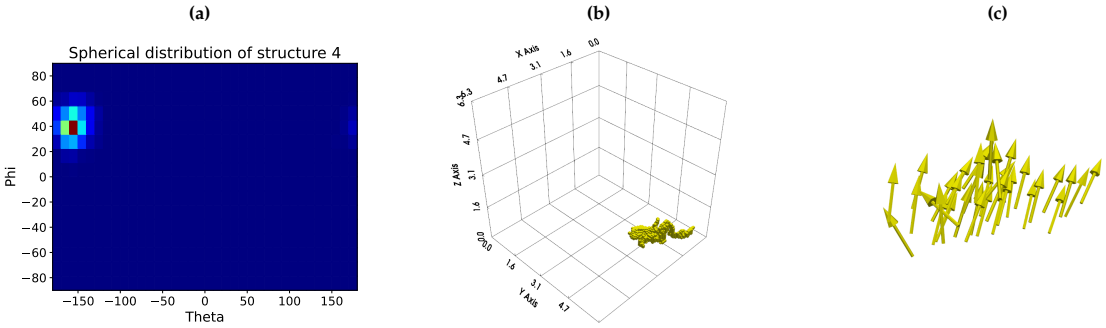


Figure A.2: Example of detected structure in the $Re_\lambda = 37$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

A.2. Reynolds number $\mathrm{Re}_\lambda \approx 65$

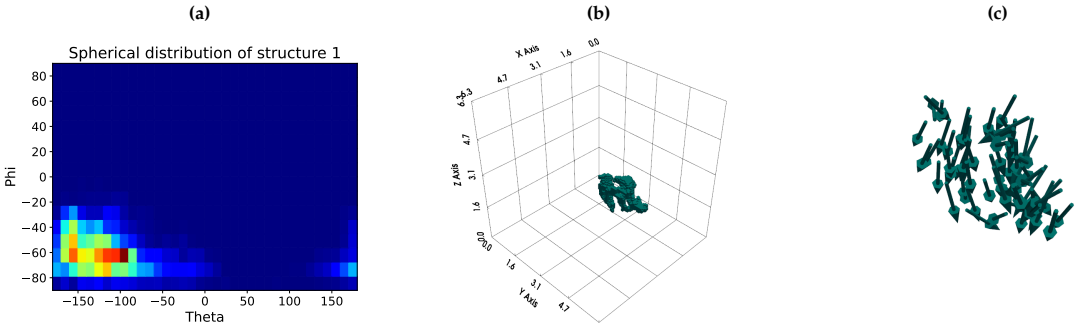


Figure A.3: Example of detected structure in the $Re_\lambda = 65$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

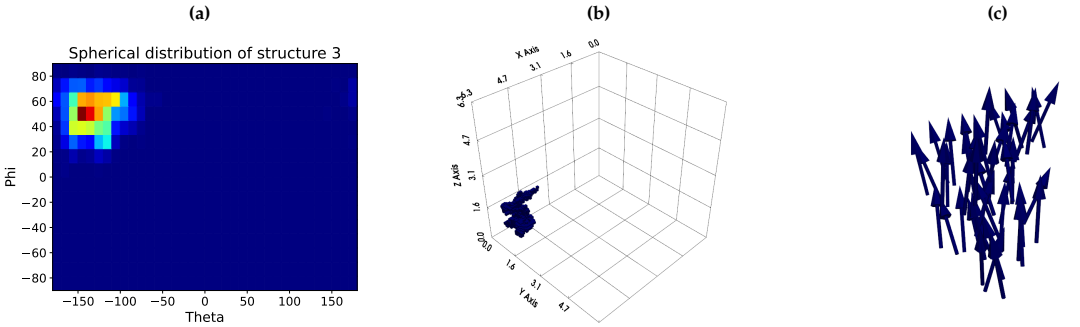


Figure A.4: Example of detected structure in the $Re_\lambda = 65$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

A.3. Reynolds number $\mathrm{Re}_\lambda \approx 97$

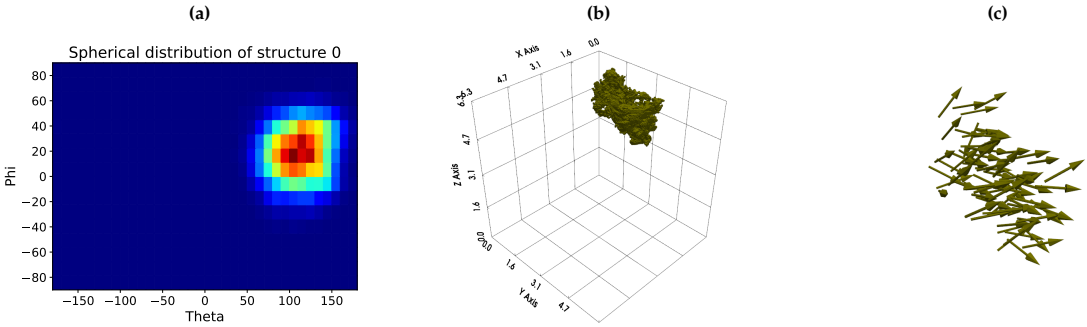


Figure A.5: Example of detected structure in the $Re_\lambda = 97$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

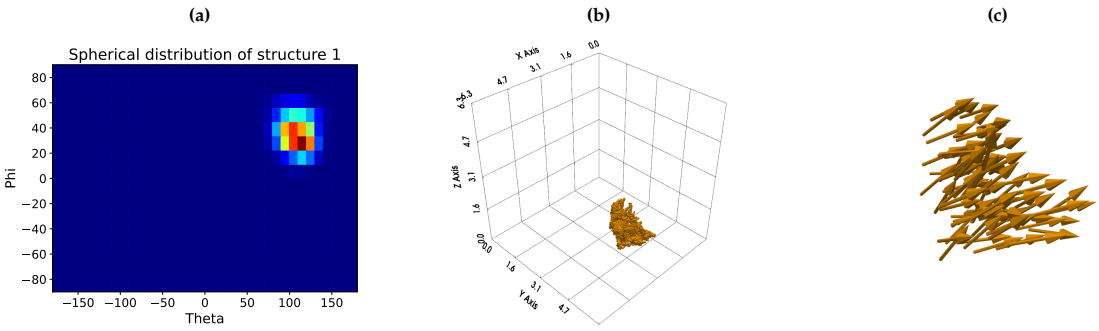


Figure A.6: Example of detected structure in the $Re_\lambda = 97$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

A.4. Reynolds number $\mathrm{Re}_\lambda \approx 141$

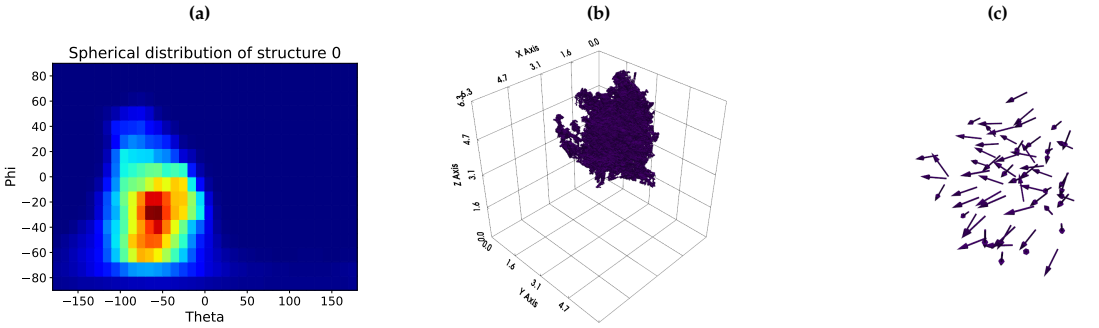


Figure A.7: Example of detected structure in the $Re_\lambda = 141$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

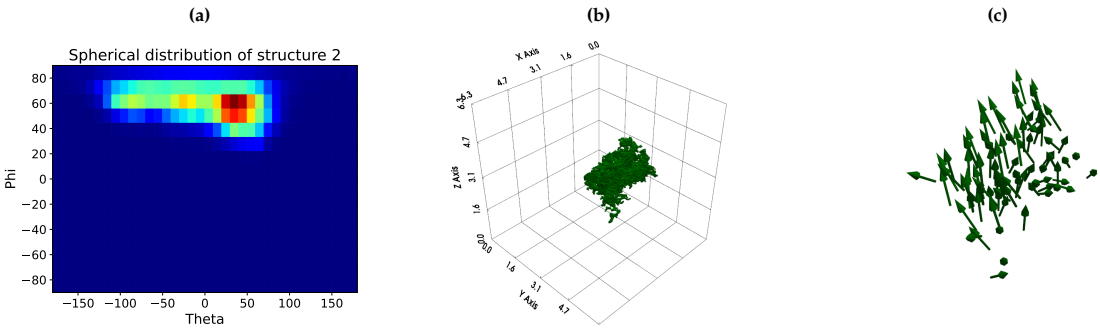


Figure A.8: Example of detected structure in the $Re_\lambda = 141$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

A.5. Reynolds number $\mathbf{Re_\lambda \approx 222}$

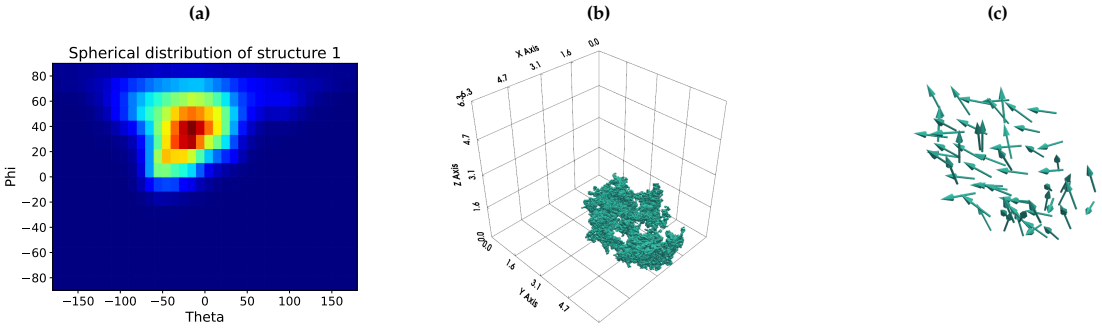


Figure A.9: Example of detected structure in the $Re_\lambda = 222$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

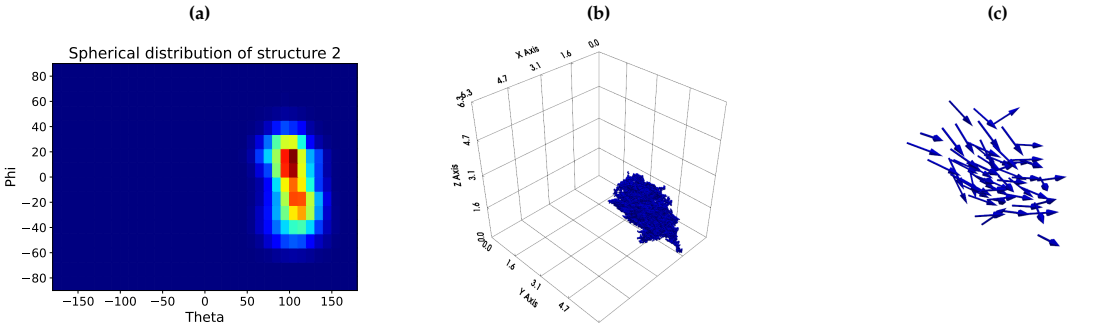


Figure A.10: Example of detected structure in the $Re_\lambda = 222$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

A.6. Reynolds number $\mathbf{Re_\lambda \approx 393}$

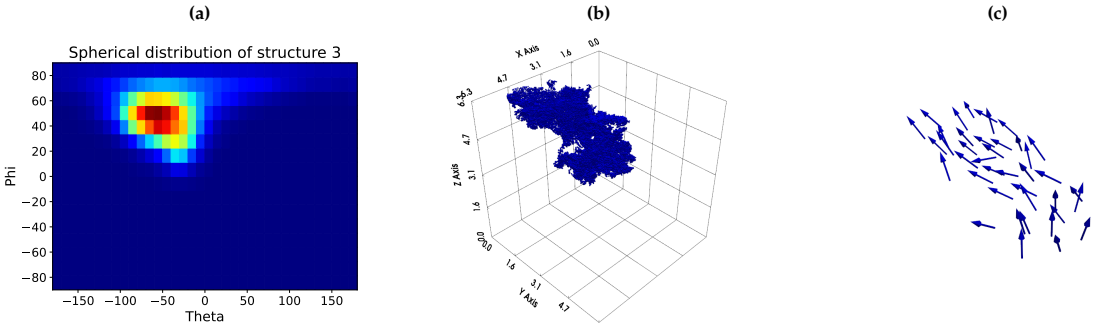


Figure A.11: Example of detected structure in the $Re_\lambda = 393$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

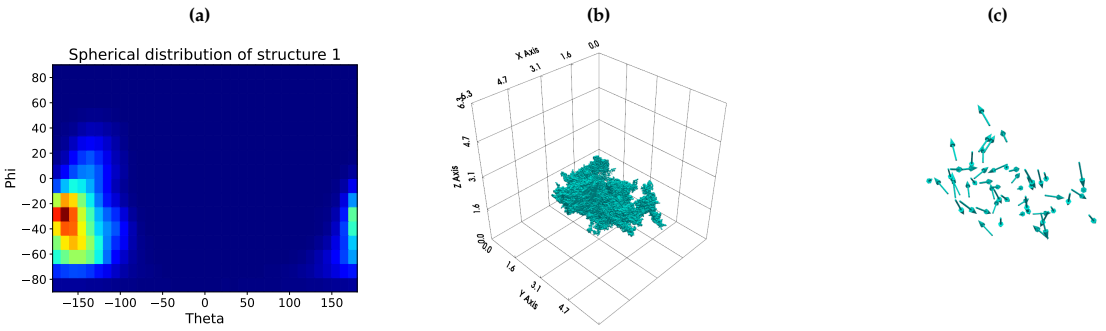


Figure A.12: Example of detected structure in the $Re_\lambda = 393$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

A.7. Reynolds number $\mathrm{Re}_\lambda \approx 433$

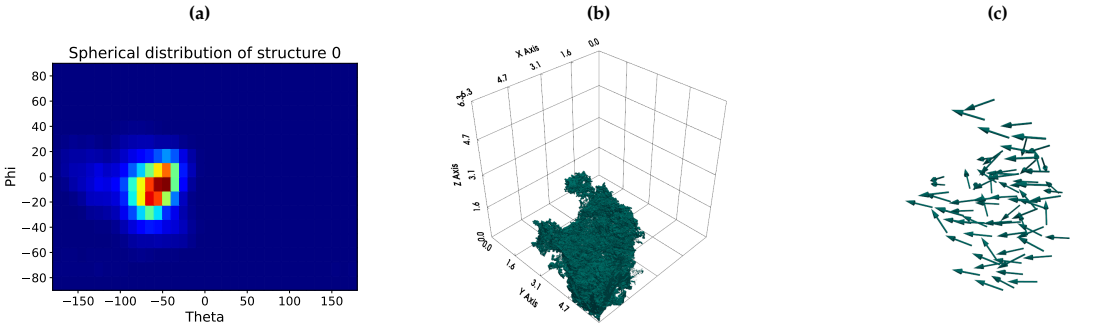


Figure A.13: Example of detected structure in the $Re_\lambda = 433$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

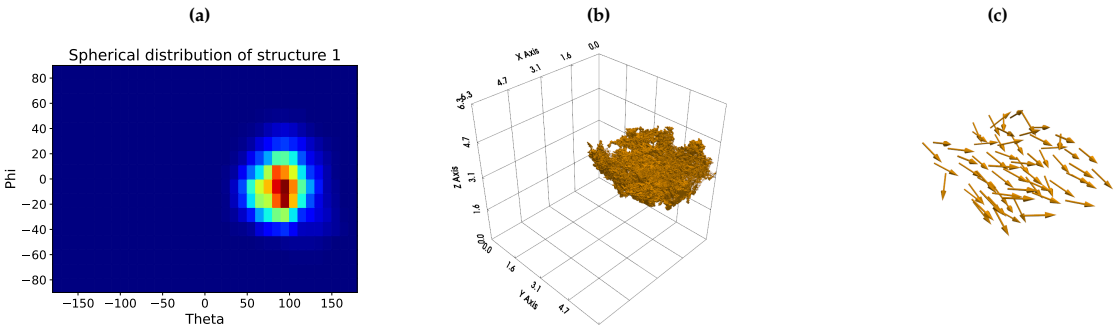


Figure A.14: Example of detected structure in the $Re_\lambda = 433$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

A.8. Reynolds number $\mathbf{Re}_\lambda \approx 730$

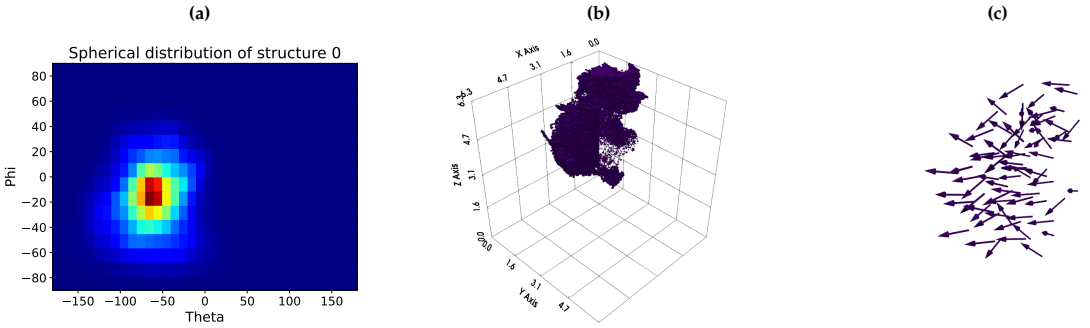


Figure A.15: Example of detected structure in the $Re_\lambda = 730$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.

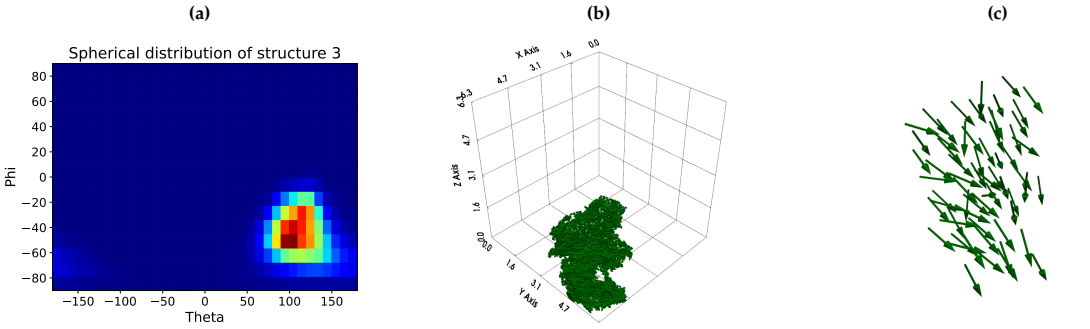


Figure A.16: Example of detected structure in the $Re_\lambda = 730$ dataset. **(a)** Spherical-angle probability density function of the structure. **(b)** Structure within the computational domain. **(c)** Velocity field inside the structure.