Data Mining Tools for the Analysis of Turbulent Flows: Needs, Issues and Potential

I. Marusic, G.V. Candler, V. Interrante, P.K. Subbareddy and A. Moss

University of Minnesota
Minneapolis, MN 55455 USA

The study of fluid flow turbulence has been an active area of research for over 100 years, mainly because of its technological importance to a vast number of applications. In recent times with the advent of supercomputers and new experimental imaging techniques, terabyte scale data sets are being generated, and hence storage as well as analysis of this data has become a major issue. In this chapter we outline a new approach to tackling these data-sets which relies on selective data storage based on real-time feature extraction and utilizing data mining tools to aid in the discovery and analysis of the data. Visualization results are presented which highlight the type and number of spatially and temporally evolving coherent features that can be extracted from the data sets as well as other high level features.

1 Introduction

Turbulence is the last great unsolved problem in classical physics, and all efforts to develop models to predict turbulent motion have fallen woefully short. The difficulty comes in because turbulent motion is a result of non-linear spatial and temporal interactions across a huge range of length and time scales.

Turbulent flows have great practical importance because they are found in a vast number of engineering applications. Examples include flow over aircraft, spacecraft, and other transport vehicles, flow inside of engines and power plant combustion systems, flow in chemical and waste processing streams, and the flow of blood in the heart and large blood vessels. In these examples, huge cost savings can be achieved if quantities like drag, combustion efficiency, and pollutant output can be predicted accurately for prototype designs. However, these quantities cannot be predicted because we have an inadequate understanding of how turbulent *structures* evolve and interact with one another.

Over the last few decades, a broad consensus has emerged that coherent or organized vortical structures (eddies, whorls, or regions of swirling flow) dominate turbulent flows and are the most important physical mechanisms for generating and sustaining turbulent motion. These structures appear in a wide variety of shapes and sizes.

The large scale eddying motions can be seen in nature, as for example, in the flow of water around obstacles, or behind your spoon when you are stirring your coffee. The smallest eddies in the flow are believed to function as sinks of energy, dissipating kinetic energy into heat through fluid friction (or viscosity). The larger scales interact with one another to spawn new larger or smaller scale eddies that cause mixing between adjacent regions of the flow.

To illustrate this interaction, consider the Earth's atmosphere where the largest scale turbulent structures are as large as the height of the clouds or the diameter of a hurricane, while the smallest eddies are about as large as the thickness of a pin. As we know from the variations in the wind strength and direction during an intense storm, the structures at various scales are not static.

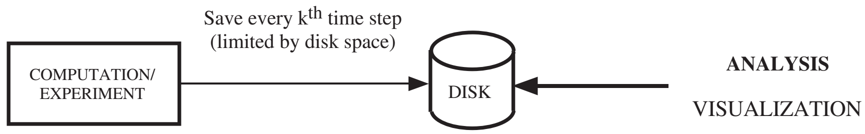

Figure 1: Schematic of a conventional data analysis approach.

The structures could be breaking up, dissipating into smaller and smaller ones or they could be coalescing, forming larger scales. The whole process is neatly paraphrased in the poem by Augustus DeMorgan:

> Great fleas have little fleas upon their backs to bite'em,
> And little fleas have lesser fleas, and so ad infinitum.
> And great fleas themselves, in turn, have greater fleas to go on;
> While these again have greater still, and greater still, and so on.

These structures are the essence of turbulence, and lacking the information to form a theoretical analysis of their formation and evolution, we are left with having to actually take a deep long look at them in circumstances we believe we control. This can be done with high-quality wind-tunnel measurements or using direct numerical simulations (DNS). These simulations must resolve all possible length and time scales, which results in extremely large calculations, involving the solution of the governing equations on multi-million point grids over many thousands of time steps. Typical calculations require a month or more on the largest parallel supercomputers and are currently limited to simplified flows at unrealistic flow conditions. (Imagine trying to resolve the motion of a large storm that has an overall dimension covering a substantial portion of a continent, while its smallest scale motion is the diameter of a pin!)

Because direct numerical simulations are very costly, an important goal of fluid dynamics researchers is to develop accurate, physically-based numerical models to predict turbulent flows. This model development depends on data from experimental measurements and numerical simulations. With the ever increasing speed of computers and the developments in advanced experimental techniques, these methods generate vast amounts of three and four dimensional data, and hence storage as well as analysis of these data has become a major issue and limitation to the extraction of relevant information.

A typical direct numerical simulation of a turbulent flow produces many terabytes of data, of which perhaps only 0.1% is stored and analyzed. For example in the turbulence simulations such as that of Martin & Candler[1], data are generated at a rate of approximately 3 terabytes per day, every day for about a month of computing. Typically, only every one-thousandth time step is actually written to disk, with the final database being approximately 2 terabytes. In particle image velocimetry (PIV) experiments data are acquired at a rate of tens of gigabytes per hour, but management and processing of an hour's worth of data typically takes many days. The data-overload situation will only worsen with further advances in computing power and experimental data acquisition hardware. Therefore, the current paradigm in scientific data acquisition and analysis, as represented in Figure 1, has become out-moded.

In this approach, existing visualization tools are used to take a series of averages in an attempt

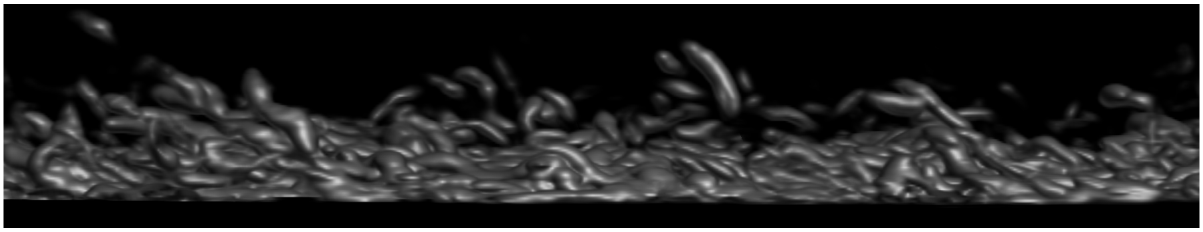

Figure 2: Coherent structures in a turbulent boundary layer flow extracted using discriminant-based structure identification. (See section 3 for details)

to uncover some type of global information. More recently movies are made of a particular flow variable (typically a scalar or vector quantity) to reach a better, though essentially qualitative, understanding of the flow physics.

Furthermore, data are stored periodically as the simulation progresses; however the interval between storage is not a phenomenological one and is usually assigned at the start of the simulation without regard to the flow evolution. Storing data at regular intervals can only capture instantaneous snap-shots of the flow and has little hope of capturing relevant time and space evolving dynamics. This is not to say that researchers are performing studies incorrectly. Rather there is presently no means of efficiently storing and analyzing the enormous amounts of data being generated. For example, the wall turbulence simulation database of Spalart (1988) is *still being analyzed twelve years after it was generated.* Several researchers are still using this data set to provide a description of wall turbulence and developing a better *feel* for turbulent fluid motion [2, 3, 4]. Figure 2 shows a typical result from such studies where vortical structures are identified in a turbulent channel flow. Such techniques have been very useful in helping to identify what structural features exist in the flow. However, without improved data analysis methods, advances in computing and data acquisition technology will not be accompanied by an improved understanding of the underlying physical processes.

In this article we wish to outline a new approach to the analysis of large scientific data sets involving dynamic and transient processes. Successful implementation will involve developing new interactive visualization and feature extraction tools and employing newly-developed scalable data mining methods.

2 New Approach for Data Analysis

Conventional data analysis and visualization separates the analysis of the data from its generation. This is inadequate for the analysis of large scientific data sets. There is simply too much data, it takes too long to make "pictures" of it, and there is too much to look at all at once. Even if it were somehow possible to save and look at everything, we might not necessarily want to. It can be argued that some of the most significant advances in understanding will come, not from showing more, but from showing less. Instead of relying on faster computers to visualize the data we advocate a new methodology, as represented in Figure 3.

The approach involves monitoring the simulation or experiment and selectively storing the data in close proximity to an event of interest – termed the "trigger." For example, while a wind tunnel is being run, the data would be streamed to a multi-processor computer where it is stored temporarily and analyzed in real time. As more data are generated, the oldest data are thrown away until an

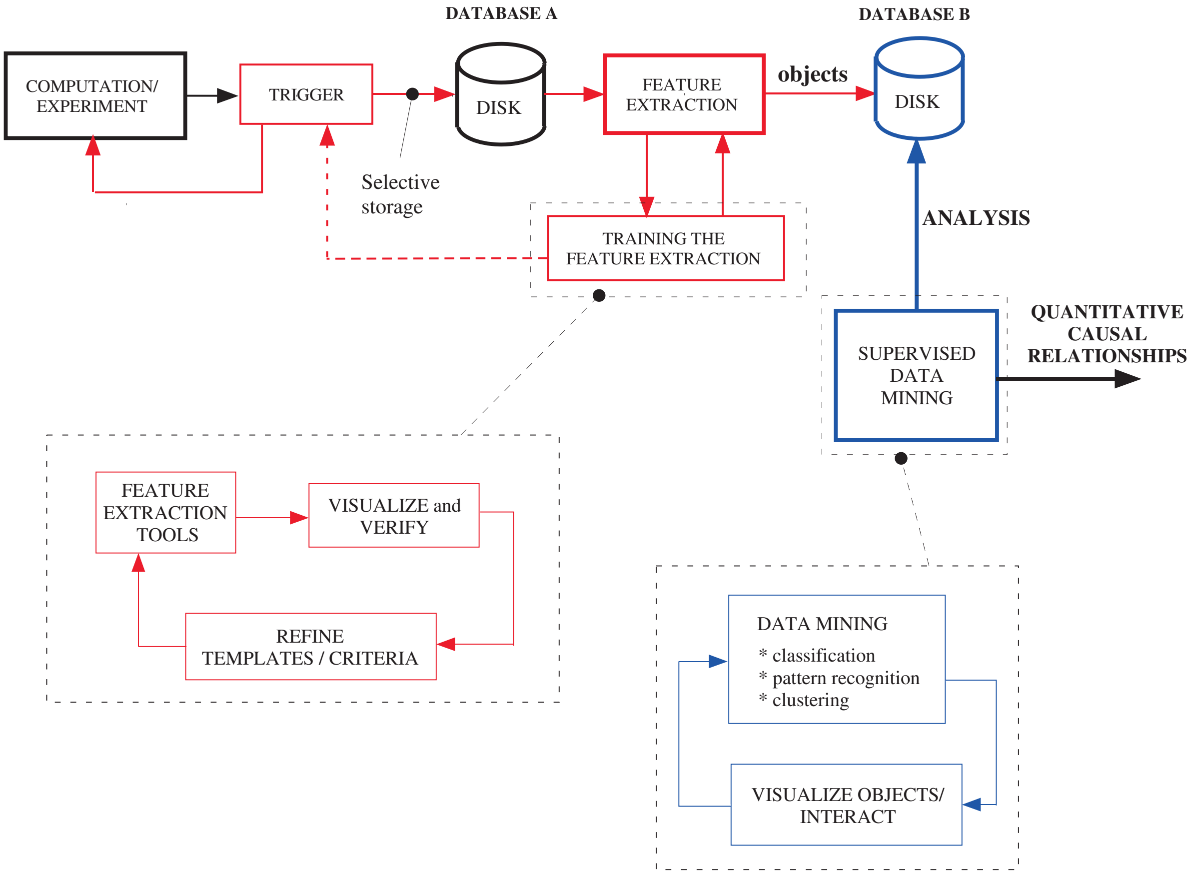

Figure 3: Schematic of our new data analysis approach.

event of interest is detected through application of advanced visualization and feature recognition techniques. Then the data remaining in memory in the spatial and temporal vicinity of the event are identified as relevant and are written to disk (Database A). A similar approach would be used to detect relevant information in a numerical simulation. In this case, data are analyzed as generated and triggers are identified on the fly. This is in contrast to the existing paradigm in which data are written to disk at arbitrary intervals at a rate predetermined by the available disk storage (see Figure 1). Using visualization and feature extraction to generate Database A is, in itself, a major challenge in the analysis of these flows.

With all the relevant data stored to disk, feature extraction and visualization techniques are then used to further analyze the data. The feature extraction module needs to be "trained" to identify and extract "objects" of interest from the database. An object quantifies the key features of each relevant turbulent structure, such as its strength, spatial extent, persistence, and location in space-time. This is where strong collaboration and interaction between the engineering and computer science/visualization experts is needed. An iterative process is required, in which the extraction criteria are refined to identify the features relating to the event. Once the module has been trained, the feature extraction can be run to produce a new compact database of objects along with their parametrizations.

This new database of objects (Database B) is then analyzed by using data mining techniques to search for patterns of interest and provide high-level information about the objects, and thus help unravel the physical mechanisms and causal relationships in the flow. Again, this requires close interactions between the engineering and computer science experts. Successful implementation of these tools would result in a powerful new means to visualize and interpret mechanisms which are currently elusive. It will allow us to pose an entirely new class of questions concerning how one event causes another in turbulent flows.

3 Case study: a Mach 4 turbulent boundary layer

Let us consider a specific turbulent flow, which is illustrated in Figure 4: the turbulent boundary layer created on the Space Shuttle during re-entry into the atmosphere. The turbulence is generated in the narrow region between the surface of the Shuttle and the high-speed external flow. In this boundary layer, there are extreme velocity and temperature gradients that cause the flow to become turbulent. The turbulent motion affects the aerodynamic forces acting on the Shuttle, but more importantly for this application, the turbulence significantly increases the aerodynamic heating of the Shuttle.

Boundary layer flows also determine the lift and drag performance of aerodynamic surfaces such as wings, propellers, and turbine blades. The turbulent motion determines if a flow will separate, which results in sudden loss of lift and substantial drag losses. A large portion of the engine power of aircraft, trains, and automobiles goes into overcoming turbulent drag losses. Boundary layers also occur on combustor surfaces and they are often the site of combustion inefficiencies and quenching that result in unwanted pollutant formation.

Visualization studies have provided a good qualitative appreciation of the structures that exist in wall turbulence and have changed the traditional notion that turbulence is a completely random phenomenon. As discussed above, it is now clear that turbulent boundary layers are largely made up of organized and quasi-deterministic, coherent vortex structures; an example is seen in Figure 2. Physical models for the kinematic state of the flow have been proposed based on these ideas using a statistical distribution of vortex structures [5, 6]. These formulations have been largely successful in reproducing second-order statistics of the turbulence, given the mean flow distribution. Unfortunately, being kinematic stochastic models, they say nothing about the dynamic interaction between the turbulent structures, and we still need to understand how these structures are generated and evolve. Thus, information on the dynamic interaction between between structures would greatly improve our understanding of turbulence.

Of particular practical interest is the phenomenon known as "bursting", which is characterized by a violent ejection of near-wall fluid into the outer part of the boundary layer, resulting in large drag and heat transfer. The frequency of bursting is directly related to the skin friction drag [7] and if we could devise schemes that reduce the average time between bursts, significant benefits would result. For example, for every 1% reduction in skin friction achieved, the US aviation industry would save an estimated $200 million per year in fuel consumption. An example of the bursting phenomenon was captured in the high Mach number boundary layer simulation by Martin & Candler [1] and is shown in Figure 5. Here representative spanwise planes (the flow is into the page) are shown at one time step and temperature is the variable plotted. We see large coherent structures that lift the high temperature near-wall fluid (white and red in Figure 5) into the cool free-stream (blue). Apart from large drag, these bursting events are responsible for increasing the heat transfer rate by a factor of about four compared to a non-turbulent laminar boundary layer. If we could understand how the events are generated, it may be possible to develop actuators to suppress or stimulate their formation, depending on whether decreased or increased heat transfer

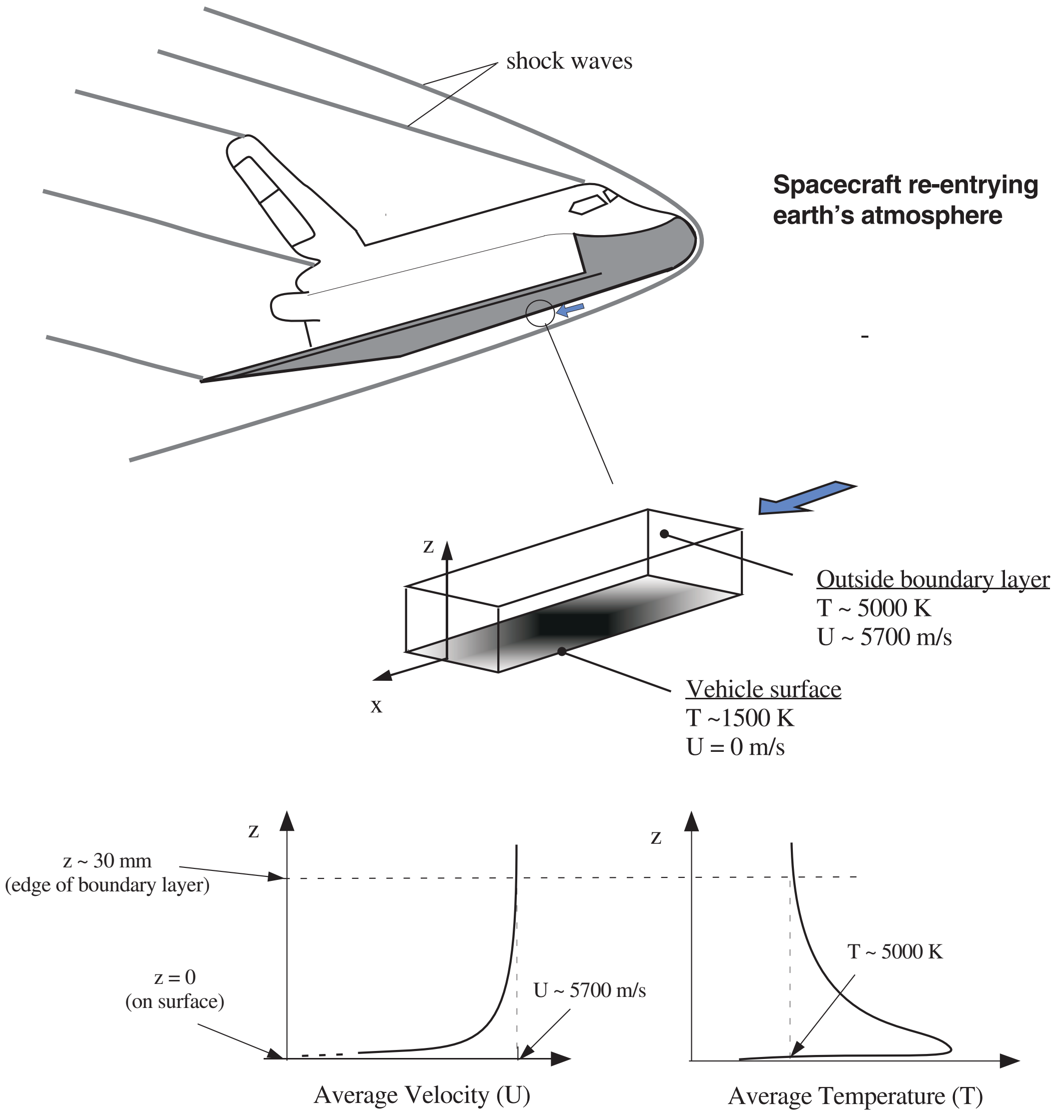

Figure 4: Illustration of a turbulent boundary layer flow on the Space Shuttle during re-entry. Because of the huge computational cost, only a small portion of the flow can be simulated; this is denoted by the boxed region.

is desired.

3.1 Details of simulation

The flow conditions for this case study are characteristic of a turbulent supersonic boundary layer at high altitude flight conditions. The data are from a large direct numerical simulation (DNS) data set of Martin & Candler [1] for a Mach 4 turbulent boundary layer with an edge temperature of 5000 K and density of $0.5\,\mathrm{kg/m^3}$. These conditions are representative of the flow sketched in Figure 4.

The DNS solves the discretized time-dependent compressible Navier-Stokes equations, which essentially represent the conservation of mass, momentum, and energy. This was done using a high-accuracy, low-dissipation, shock-capturing finite difference method specifically designed for

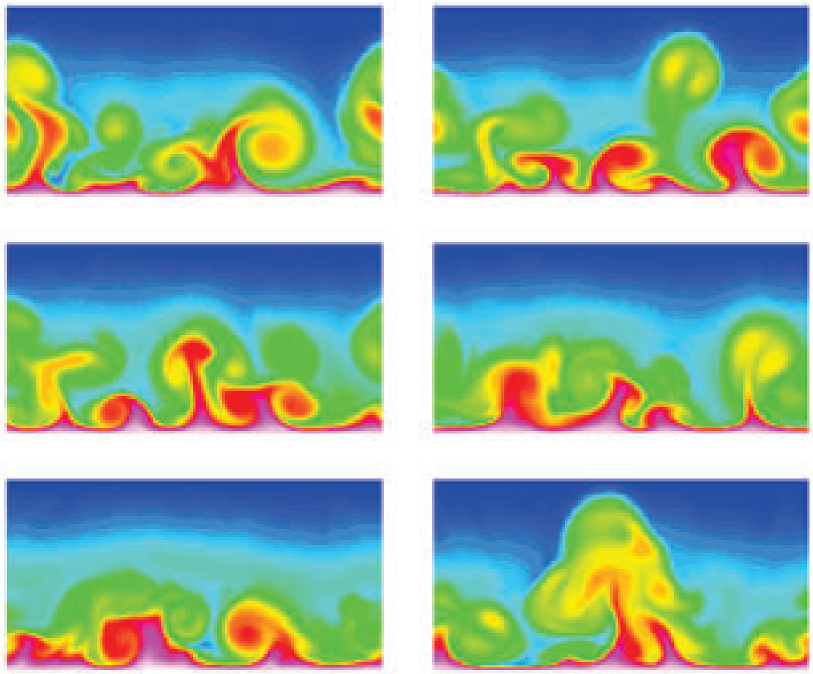

Figure 5: Example of "bursting" shown in spanwise planes from the Mach 4.0 adiabatic wall boundary layer simulation of Martin & Candler [1].

this type of problem [8]. The time integration uses a parallelizable second-order accurate implicit method [9].

The grid in the streamwise and spanwise directions is equispaced and the wall-normal direction is geometrically stretched from the wall. The domain size is measured in units of the boundary layer thickness δ, and for this case is $8\delta \times 2\delta \times 15\delta$ [1]. The grid resolution is $\Delta x^+ = 15$, where the superscript denotes normalization using the viscous length scale (ν/U_τ), $\Delta y^+ = 11$, with Δz^+ varying from 0.15 to 46 across the boundary layer. To achieve adequate resolution for this domain size, the DNS was performed on a grid with $384 \times 128 \times 128$ points in the streamwise, spanwise, and wall normal directions, respectively.

The calculations were performed using a 256 processor Cray T3E at the University of Minnesota Army High Performance Computing Research Center (AHPCRC). There are no detailed experimental data for these flow conditions, but the results of the DNS agree very well with theoretical and experimental correlations. It should be noted that the DNS results are statistically exact and any variable of interest can be constructed from the solution at any given time.

3.2 Identification of structures

The structures that form the basis of turbulent motion are believed to be composed of regions of high vorticity, or angular velocity, which very roughly speaking means that the fluid particles are spinning around themselves and other fluid particles. The picture is confused however, by the fact that there are regions of flow with high vorticity, but are not part of what are believed to be coherent structures.

In order to separate the spurious vorticity from that within the structures of interest, we must consider an appropriate frame of reference. For example, the wake behind a boat appears to be highly disorganized when seen from the shore, but if one is on the boat, the wake assumes an organized structure. Similarly, if we move with a particle of fluid in the flow, it becomes clear whether the fluid is swirling about that point and is part of a coherent vortical structure.

It is also useful to consider the local streamline pattern at a point, which is computed using the gradients of each of the components of the velocity. This velocity gradient tensor ($\partial u_i / \partial x_j$, where i and j run over all three of the spatial dimensions) contains all nine possible velocity derivatives. Visualizing a field made up of nine quantities at each point is usually a futile exercise, and we must distill one number out of this matrix that quantifies the swirling vortical motion. Quite naturally, we are led to the eigenvalue problem for the matrix. Being a 3×3 matrix, it has 3 eigenvalues which are either all real or have a pair of complex roots and one real root. A little analysis shows that the latter case is when the local streamlines exhibit the swirling pattern that we were looking for, and in fact, the complex part of the eigenvalues indicates the strength of the swirl. The number that determines whether there are real or complex roots is the discriminant (D) of the characteristic equation of the eigenvalue problem. If $D > 0$, we have a swirling flow, and a connected region of positive values of D would be an unambiguous definition for a coherent structure. It is this parameter, D, that we currently use to identify the structures in the flow.

3.3 Visualization tools and results

The DNS data are rendered using a classical ray-casting volume renderer (Levoy [10]). At each time step, the 3D normalized level-set data are first resampled into a high resolution uniform grid ($1018 \times 251 \times 137$), and then to minimize disk space and memory requirements quantized into 8-bit values spanning the range 0-255, such that the remapping is uniform across all time steps. Images are created by tracing a bundle of rays from the eye point through the 3D data, such that each ray passes through the center of a pixel in the image plane, and returns the color value to be assigned to that pixel. The final pixel color is obtained by compositing local color and opacity values trilinearly interpolated from the uniform grid to sample points at evenly spaced intervals along the ray.

Local color values are computed at each grid point using a simple two-sided Phong illumination model (Foley *et al.* [11]):

$$\text{sample\_color} = k_a + k_d \times |N \cdot L| + k_s \times |N \cdot H|^n,$$

where k_a, k_d and k_s are the ambient, diffuse and specular reflectivities multiplied by the intensity of illuminant, respectively; which together with the exponent n are defined by the user. N is a unit vector in the direction of the data value gradient, which can be interpreted as the unit normal to the level surface at the grid point, L is the direction of the parallel illumination from an infinitely distant light source, and H is the vector halfway between L and the line of sight, or direction of projection. Local opacity values are defined at each grid point as a gradient magnitude-weighted linear function of the input data values according to a user-specified data-to-opacity correspondence given at key breakpoints. The volume rendering approach in effect finesses the definition of the apparent surfaces, allowing greater flexibility to portray the structures of interest in a form that is robust to the local effects of isolated samples and minor fluctuations or nonuniformities in the distribution of the chosen level-set. Our experience shows that this volume rendering yields more clearly connected structures than the usual iso-surface rendering. Examples of a volume rendering using the discriminant, D, to specify the illuminant for one time step from the DNS are shown in Figures 2 and 6.

In order to isolate a particular structure from the forest of other structures in the flow field, we use a volume filling method. A seed is placed within a structure of interest that is usually identified from an image such as shown in Figures 6 and 7. Then a nearest-neighbor search is done to find all connected voxels where the variable of interest (D here) is greater than a user-specified value. This connected region is then visualized as a colored rendering. A more sophisticated volume filling method that uses gradient information may result in more consistent connectivity of structures; this extension is being pursued in continuing work.

The volume filling approach can be used to construct a database of relevant objects for data mining analysis. For example, it would be straight-forward to identify all locations within a flow domain where the discriminant is above a certain value; these locations are used as seeds for the volume filling. The structures are then filled and any duplicate structures are eliminated. This results in a finite number of connected regions; the number of which depends on the specified seed and connectivity thresholds. Each structure can be parameterized by its location, size, volume-averaged swirl strength, or many other possible variables.

We would like to determine how these structures evolve in time. Figure 7 shows a typical evolution of one large-sized volume-filled structure over approximately 90 time-steps in the simulation. Since the structures move long with the flow, we can estimate to high accuracy where they will be located at the next time step. The volume filling can then be repeated, but with a very good estimate of the location and extent of the connected regions. Thus, once we have performed an initial volume filling and structure identification, updating the database of structure connectivity and parameters is straight-forward. The only concern with this approach is that entirely new structures can be born, or two structures can merge. Appropriate logic must be included to allow these processes to occur. However from our current visualizations, the frequency of large structure birth and merging appears to be quite low (this is not true for the smaller structures).

In any case, the volume filling can be used to construct a database of objects of interest. The database will contain a space-time trajectory of each structure, including all relevant parameters. Because each entry in the database represents a large ensemble of relevant voxels, it will be very significantly smaller than the raw data. For example, a representative number of large-scale structures may be several hundred; if each structure is parameterized by 10 quantities, a single time step will require only on the order of 10^4 real variables per time step. This is contrasted with the 3×10^7 real variables required to describe the entire domain. This represents a compression of a factor of 10^3; but more importantly, the small database contains only the *relevant* data.

4 Data Mining Issues

A primary issue with mining of the turbulent flow data set is the interface between the fluid dynamicists and the data miners. Fluids investigators would like to be able to ask quite abstract questions, such as "What causes a burst event?" or "Do structures group themselves together, or are they random?" These questions need to be converted into a form that is appropriate for data mining, and of course the compact database discussed above must contain the relevant information to answer a given question. Likewise, it is important for the data mining investigators to make clear what classes of questions data mining algorithms can actually answer. Thus, once we have crossed the hurdle of generating a compact data set, probably the main issue is a matter of communication

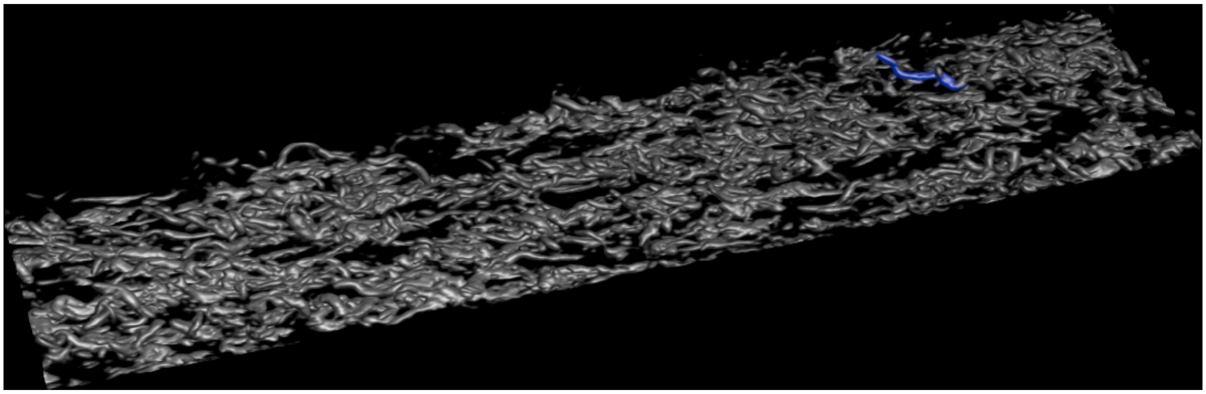

Figure 6: An isometric view of the boundary layer from the Mach 4.0 adiabatic wall simulation.

between the fluid dynamicists and the data miners.

Another issue is that the turbulence is extremely complicated with a very large number of vortical structures across a very large range of length scales, as seen in Figure 6. This complexity in the flow field results in having to consider many possible mechanisms to explain a given event or process. For example, consider the problem of understanding burst formation as indicated in Figure 5. It is expected that to predict burst formation in boundary layer turbulent flow, a model will involve both physical properties (temperature, pressure, vorticity, etc.), as well as spatial and temporal relations between the different vortical structures in the flow. Unfortunately, with our present understanding of turbulence, we do not know what these relations may be, and there is a very large number of possible correlations between these variables and their history. For example there are a number of higher level characteristics which are speculated to cause burst formation. These include the organization of vortices into coherent spatial clusters and spatial coherence among vortices or clusters of vortices [12, 13, 14]. Thus, models discovered by data mining techniques need to be able to express such relations.

As an initial approach, *training sets* need to be established. In a steady-state turbulent flow, bursts occur at unpredictable times during a simulation. Using the intelligent data filtering techniques discussed above, we can obtain objects describing the state of the flow at a time interval around the occurrence of each burst. These observations will become the *positive* instances in our training set. At the same time, we can construct a large number of objects describing the state of the flow during time intervals in which no bursts occur; thiese become the *negative* instances in our training set. From this training set, we can begin developing explanatory models that will yield quantitative insight into burst formation. These models can then be evaluated using simulation data that has not been used in the training of the model.

Unfortunately, even though the feature extraction techniques can provide us with higher order objects rather than just raw flow information, the information content of these objects is still too low to allow us to express the higher order relations that are needed. Thus, the need exists for data mining techniques based on clustering and pattern discovery that will analyze the low level objects and extract higher order features present in the data set that will aid in the development of explanatory models for the phenomena of interest.

There are several characteristics of the objects extracted from scientific data sets that makes clustering and pattern discovery challenging. Irrespective of the sophistication of the feature extraction tools, the objects extracted from the raw flow information will unavoidably have a certain

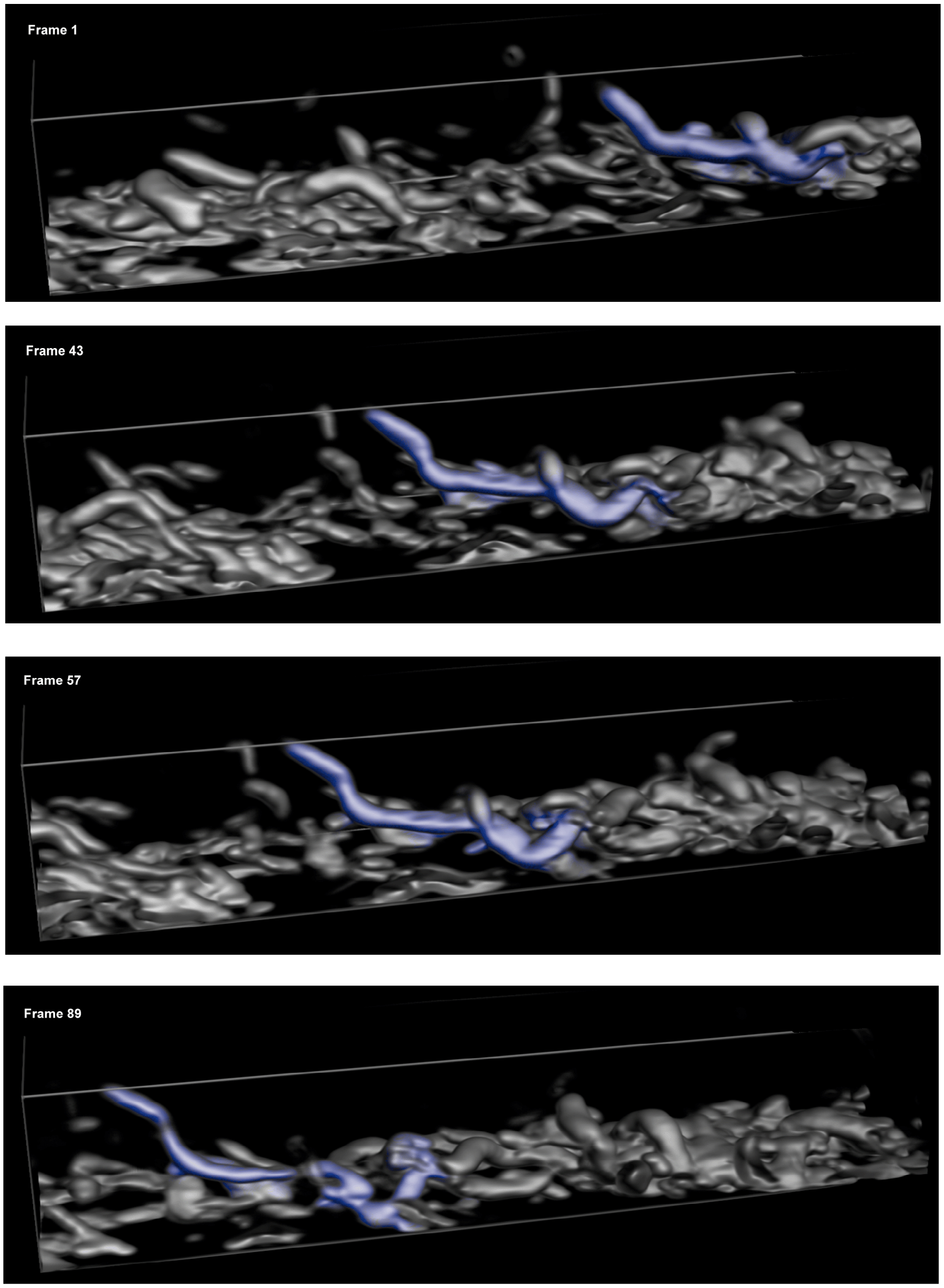

Figure 7: Figures above show the time evolution of a typical structure seen in the simulation.

degree of noise and imprecision. Similarly, the number of objects in the database can easily run to several thousands and increases as the simulation/experiment duration increases. At the same time, the parametric description of these objects can easily involve tens of features, and since the objects arise in a three dimensional time varying simulation, they have spatial as well as temporal features. Existing data mining techniques are not well suited to operate on data sets that combine the above characteristics.(REFERENCES?)

Developing clustering algorithms for scientific data sets poses a number of challenges due to both the characteristics of the data itself as well as the types of the desired clusters. The clusters may be of variable sizes and densities, and may be of arbitrary shapes. The need exists to develop clustering algorithms that are capable of accommodating different clustering objectives. For example, the problem of finding a cluster that is a packets of vortices is entirely different than that of finding a cluster that contains vortices of the same size. (REFERENCES?)

In the context of pattern discovery algorithms, the patterns to be considered here involve spatial, temporal, or spatio-temporal relations among objects. For example, interesting patterns may involve vortices oriented in a certain way, or a certain type of vortices formed after another type, or a set of small vortices evolving to create larger vortices.

The problem of finding these patterns becomes even more challenging when the objects are higher order features such as clusters. In this case establishing relationship among clusters from different events itself can be non-trivial. The need therefore exists to extend existing pattern discovery algorithms by allowing them to discover arbitrarily complex patterns that represent the relationships among events. (REFERENCE to other DM chapter)

5 Acknowledgments

We would like to thank Vipin Kumar, George Karypis and Eui-Hong (Sam) Han for sharing with us their insights into Data Mining. This work is made possible by the support of the National Science Foundation through grant ACI-9982274, and this is gratefully acknowledged.

References

[1] M. P. Martin and G. V. Candler. DNS of a Mach 4 boundary layer with chemical reactions. *AIAA-2000-0399*, 2000.

[2] S. K. Robinson. Coherent motions in turbulent boundary layers. *Annu. Rev. Fluid Mech.*, 23:601–639, 1991.

[3] B. J Cantwell, J. M. Chacin, and P. Bradshaw. On the dynamics of turbulent boundary layers. In *Self-Sustaining Mechanisms of Wall Turbulence*, Ed. R.L. Panton, Comp. Mech. Publications, 1997.

[4] J. M. Chacin and B. J Cantwell. Dynamics of a low Reynolds number turbulent boundary layer. *J. Fluid Mech.*, 404:87–115, 2000.

[5] A. E. Perry and I. Marusic. A wall-wake model for the turbulence structure of boundary layers. Part 1. Extension of the attached eddy hypothesis. *J. Fluid Mech.*, 298:361–388, 1995.

[6] I. Marusic and A. E. Perry. A wall-wake model for the turbulence structure of boundary layers. Part 2. Further experimental support. *J. Fluid Mech.*, 298:389–407, 1995.

[7] J.L . Lumley and P. Blossey. Control of turbulence. *Annu. Rev. Fluid Mech.*, 30:311–327, 1998.

[8] V. G. Weirs and G. V. Candler. Optimization of weighted ENO schemes for DNS of compressible turbulence. *AIAA-97-1940*, 1997.

[9] D. Olejniczak and G. V. Candler. Numerical testing of a data-parallel LU relaxation method for compressible DNS. *AIAA-97-2133*, 1997.

[10] M. Levoy. Display of surfaces from volume data. *IEEE Comp. Graphics App.*, 8(3):29–37, 1988.

[11] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes. *Computer Graphics: Principles and Practice.* 2nd ed., Addison Wesley, 1990.

[12] R. J. Adrian, C. D. Meinhart, and C. D. Tomkins. Vortex organization in the outer region of the turbulent boundary layer. *J. Fluid Mech.*, 422:1–53, 2000.

[13] J. Zhou, R. J. Adrian, S. Balachandar, and T. M. Kendall. Mechanisms for generating coherent packets of hairpin vortices in channel flow. *J. Fluid Mech.*, 387:353–396, 1999.

[14] I. Marusic. On the role of large-scale structures in wall turbulence. *Phys. Fluids*, 2001. In Press.