Book Chapter Template

From Data Chaos to the Visualization Cosmos

Chao Tong and Robert S. Laramee, Visual And Interactive Computing Group, Swansea University, Swansea, UK, {806708, r.s.laramee}@swansea.ac.uk

Abstract

Data visualization is a general term that describes any effort to help people enhance their understanding of data by placing it in a visual context. We present a ubiquitous pattern of knowledge evolution that the collective digital society is experiencing. It starts with a challenge or goal in the real world. When implementing a real-world solution, we often run into barriers. Creating a digital solution to an analogue problem create massive amounts of data. Visualization is a key technology to extract meaning from large data sets.

Keywords: Data Visualization, Real world challenge, Data chaos, digital solution

1. Introduction

Data visualization is a general term that describes any effort to help people enhance their understanding of data by placing it in a visual context. Patterns, trends and correlations that might go undetected in numeric on text-based data can be exposed and recognized easier with data visualization software [1]. Figure 1 represents a ubiquitous pattern of knowledge evolution that the collective digital society is experiencing. It consists of six basic constituents. It starts with a challenge or goal in the real world. The goal could be to build or optimize a design, like a car or computer. The start could be a challenge such as reaching a new level of understanding or observing a behavior or phenomenon rarely or never seen previously. The goal could be running a successful business and making a profit. We all have real world goals and challenges. We all have new understanding and knowledge we would like to obtain. We all have things we would like to build, create, and optimize.

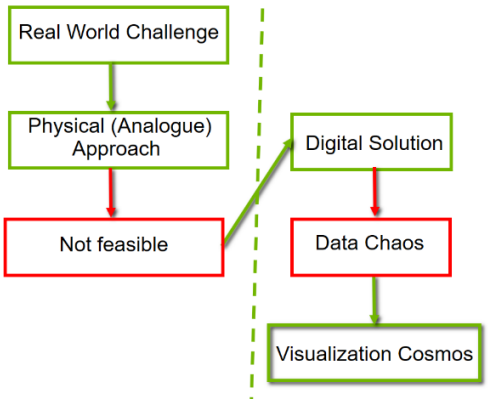

Figure 1 :A ubiquitous pattern of knowledge evolution. Here we present the model of our collective thought process.

When trying to build something we generally know that whatever it is, it can theoretically be built in the real-world. For example, cars and structures can be built out of raw materials and components with the right tools. We also know that observations can be made, in general, by being in the right place at the right time, either personally or with recording equipment. Experiments can generally be conducted with the appropriate equipment. New levels of understanding can generally be obtained if we hire enough of the right people.

However, when implementing a real-world solution, we often run into barriers. Cars and structures are extremely expensive to build and may also require a long-term investment. Observations may be very expensive, very difficult, or even impossible. Some observations interfere with the very behavior or phenomena they are trying to study. Recording equipment may be too expensive or cause logistical problems. Equipment for experiments is generally very expensive. This is especially true if the equipment is specialized or for very small or very large-scale investigations. Also, hiring people for new understanding may not be feasible due to expense. A full-time research assistant costs 100K GBP per year under current funding agency full economic costing (FEC) requirements in the UK. Real-world solutions are generally very expensive or not feasible at all. Some real-world solutions are impossible.

It is because of the high cost of real-world solutions that collectively, as a society, we turn to digital solutions to address our challenges and goals. The dotted line in Figure 1 separates the real, physical, or analogue world on the left side from the digital world on the right. We all look to the digital world for the answers to our questions. “There must be an app for that.” or “What app can be built to solve this problem?” is the collective thinking in this day in age. Society looks towards digital solutions for their real-world problems to deliver the user from the dilemma they may face. People believe that software is less-expensive to build than objects in the real world. The virtual world should be more feasible than the physical or analogue world. And this is true in many scenarios.

However, creating a digital solution to an analogue problem introduces new challenges. In particular, digital solutions, including software, create massive

amounts of data. The amount of data digital approaches generate is generally unbounded. Software and storage hardware is less and less expensive with time. Thus, users collect, collect, and collect even more data. This is the point at which the knowledge evolution pipeline of Figure 1 becomes interesting. Large collections of complex data are not automatically useful. Extracting meaningful information, knowledge, and ultimately wisdom from large data collection is the main challenge facing the digital world today. The collection of essentially unbounded data is what we term data chaos. Collecting and archiving data without careful planning and well thought out information design quickly or slowly results in a chaotic data environment. Those who collect data are generally not yet aware of how difficult it is to then derive useful insight and knowledge from it.

On the other hand, the knowledge that visualization is a key technology to extract meaning from large data sets is rapidly spreading. This is one solution to the data chaos. In the early years of data visualization as a field, say the first 10 years, from 1987-1997, data visualization was considered very niche. Not many people knew about it nor knew of its existence. It is only since around the turn of the century that word started to spread. In the 2000s the first main-stream news stories including the phrase 'Data Visualization' were published. Nowadays, the field has come a long way from obscurity to breaking into the main stream. Its presence and importance as a field is starting to become understood. Word is spreading that a data visualization community exists and that this is a topic a student can study at university.

That's the basic pattern of knowledge evolution. The rest of the chapter provides concrete examples of these six stages from real-world challenges to the visualization cosmos. The focus is on the last two stages: from data chaos to the visualization cosmos.

2. The Universal Big Data Story (and Quandary)

We can find this pattern everywhere. It doesn't matter where we look. We can see in computational fluid dynamics. Physicists and astronomers are facing these dilemmas. It's not possible to study all the stars and black holes physically. We see this pattern with marine biologists, biochemists, psychologists, sociologists, sport scientists, journalists, and those studying the humanities. We see this evolution with government councils, banks, call centers, retail websites, transportation. The list is virtually endless. You can experience this yourself as you collect your own photos. People like to collect things. This is another contributing factor to the data chaos. A person may not even have a goal to reach or a problem they are trying to solve. They just like to collect.

3. The Visual Cortex

Data visualization uses computer graphics to generate images of complex data sets. It's different from computer graphics. "Computer graphics is a branch of computer science, yes, but its appeal reaches far beyond that relatively specialized field. In its short lifetime, computer graphics has attracted some of the most creative people in the work to its fold," from the classic textbook "Introduction to Computer Graphics" by Foley et al, 2000. Visualization tries to generate images of reality.

Visualization exploits our powerful visual system. We have several billion neurons dedicated to our visual processing and visual cortex [2]. See Figure 2.

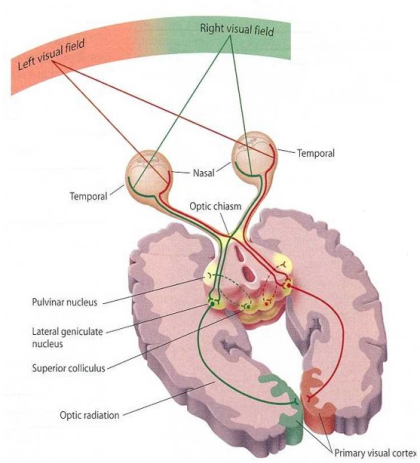

Figure 2: Several billion neurons are devoted to analyzing visual information.

The numbers of neurons are not very meaningful unless we put them into context. We have eight percent of the cortex dedicated touch and three percent dedicated to hearing. We have anywhere from 4 to 10 times of our cortex dedicated to visual processing than the other senses. It makes sense to explore the visual processing power in our brains as opposed to the other senses. It's dedicated to processing color, motion, texture, and shape.

4. Visualization Goals

Data visualization has some strengths and goals itself. One of the goals of data visualization is exploring data. This may be the case when the user does not know anything about their data set. They just want to find out what it looks like and its characteristics.

Users search for trends or patterns in the data. Exploration is for the user that's not very familiar with the dataset. Visualization is also good for analysis: to confirm or refute a hypothesis. An expert may have collected the data for a special purpose and would like to confirm or refute a hypothesis or answer a specific question. Visualization is also effective for presentation.

When our exploration and analysis is finished we can present the results to a wider audience. Visualization is also good for acceleration i.e. to speed up something such as a search process. This is often a decision-making process or knowledge discovery process. We can see things that were otherwise impossible.

5. Example: Visualization of Call Center Data

Let's look at this first example of this pattern of knowledge evolution. This is from a business context. One of Swansea University's industry collaborators is called QPC Ltd. They are an innovator in call center technology. Their goal is to understand call center behavior and to increase understanding of calls and all the activities that occur inside a call center. The call centers are staffed with many agents and the agents are

answering hundreds of thousands of calls every day. How can we increase our understanding of all those events and what is happening inside of a call center?

We theoretically could go down the analog or physical route. We could hire more people that stand and observe what's happening in the call center and attempt to take notes to enhance understanding. Or maybe CCTV could be used to try to film everything that's going on. These analogue solutions will be very expensive and not very practical. The analog solution to hire more people for just observation is not practically feasible and will cost too much money.

So QPC Ltd chose the digital solution. They decided to implement an event database. The database logs all events in the call center: who called, when they call, how much time they spend navigating menus inside the interactive voice recognition system (IVR), how long they spent in the queue before speaking to an agent, whether or not they abandon their call, which agent they spoke to, and how long they spoke to each agent etc. That digital solution in the form of a database stores of millions events everyday. A call center generates lots of activities. The UK employs over a million people in call centers or about five percent of its workforce are employed in call centers [3]. It's a large market.

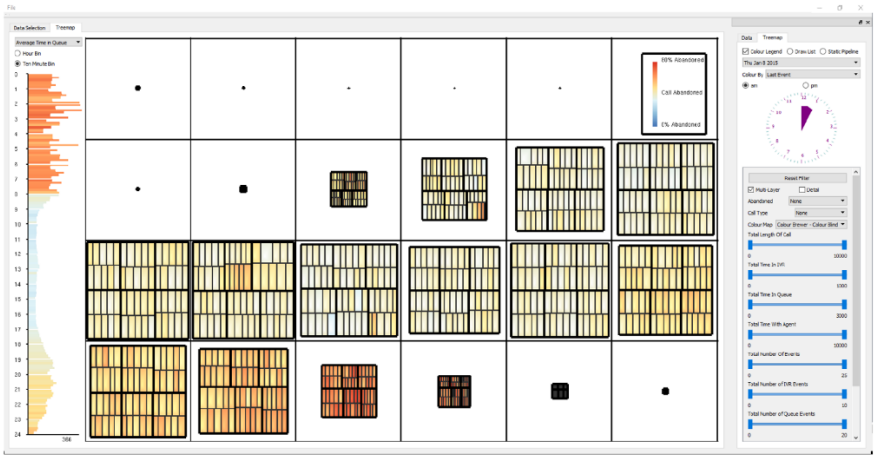

Figure 3: Visualization of call center data. Image courtesy of Roberts et al. [3].

How do we take the chaos of call center data and visualize it to make sense of it? We can use a treemap as one of the ways to visualize call center events. See Figure 3. The treemap is a hierarchical data structure. We start with an overview of the data and then zoom in down to different levels of detail. In this case, the size of the rectangles is initially mapped to call volume. The different hours start from midnight to midnight again. We can see when the call center opens and when the call volume increases and reaches its maximum at around lunchtime. Then it starts to descend again.

Color is mapped to the percentage of abandoned calls by default. We can notice call centers trying to avoid abandoned calls. We can observe a big increase in abandoned calls in the evening right after dinner around 7pm-8pm. The user can map the calls to different colors at different costs. They can also map the colors to different kinds of events for example abandoned calls or successful calls.

We can also navigate the treemap. We can zoom in smoothly and see more details. We can zoom in to single hour and each rectangle represents a single call. We can

visualize individual calls and how long they take. There is a call that lasted two hours. The unusual calls that last long time jump right out. Probably they spent a long time with an agent —a very dedicated agent spent a long time trying to solve a customer problem. The users can use a clock interface to smoothly zoom and navigate each hour. The software features a smooth zooming and panning operation and with the clock showing. The user does not get lost.

We can easily see which hours we are observing even when we zoom in. We can zoom in even further, one hour is broken up into 10-minute intervals and then those 10-minute intervals are broken up into single minute intervals. We also see a standard histogram on the left which represents the data and provides an overview. Each bar represents a 10-minute interval. Color is mapped to some data attribute chosen by the user in this case the average call length which we can see up in Figure 3. We can see, suddenly during, the evening average call length increases, and we can see over the day the average call length increases throughout the day as an overall trend.

The treemap features a fine level of detail. Each rectangle can represent a single phone call and, in this case, how long each call lasted. At the top level are not individual calls. Each rectangle represents an hour and then each hour is broken up into 10-minute blocks. So we have 6, 10 minute blocks and then each time in the block is broken up into individual minutes. This is an exciting project because this is the first time that QPC Ltd have ever seen overview of the call center activity in any way shape or form. As soon as we see the overview we can easily make observations about the call center volume about the increasing level of abandon calls. The average call length is also increasing as we examine the day.

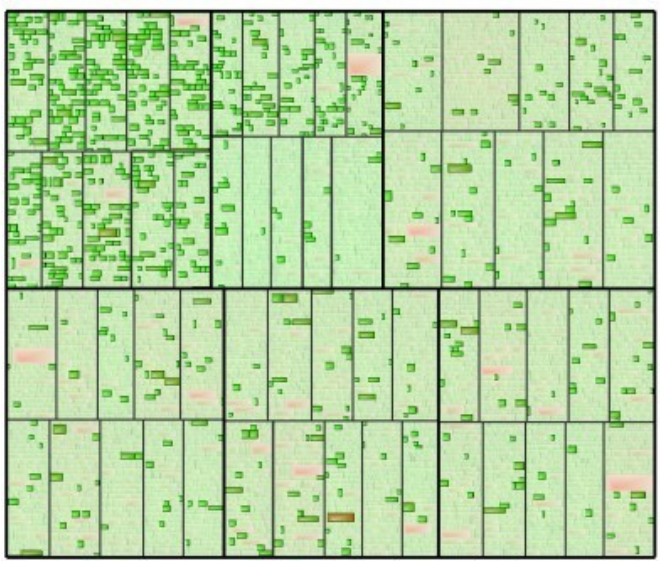

Figure 4: Focus + context filtering feature of call center data. Image courtesy of Roberts et al. [3]

We can filter calls using different sliders. This is the analytical part of the process. This is an example of focus and context visualization. See Figure 4. We focus on the calls that spend a longer time in a queue.

We can focus on the inbound calls because call centers have inbound calls and outbound calls. These can be filtered by completed calls. We can combine filters in different ways.

We can click on an individual call and then obtain the most detailed level of information like how much time the caller spent in the IVR navigating menus, how much time they spent queuing and how much time they spent talking to agents. We have two different queuing events, an agent event, a second agent event, back in the queue, back to another agent, back into the queue again. That is a complicated phone call. That is the lowest level of detail. We can also see the type of call in this case a consult call as it shows the number of events, one IVR event for queuing events and four different agent event.

One detailed view shows that the proportion each event as a unit proportion because sometimes the events disappear when they're too short for a traditional version.

6. Example: Investigating Swirl and Tumble Flow with a Comparison of Techniques

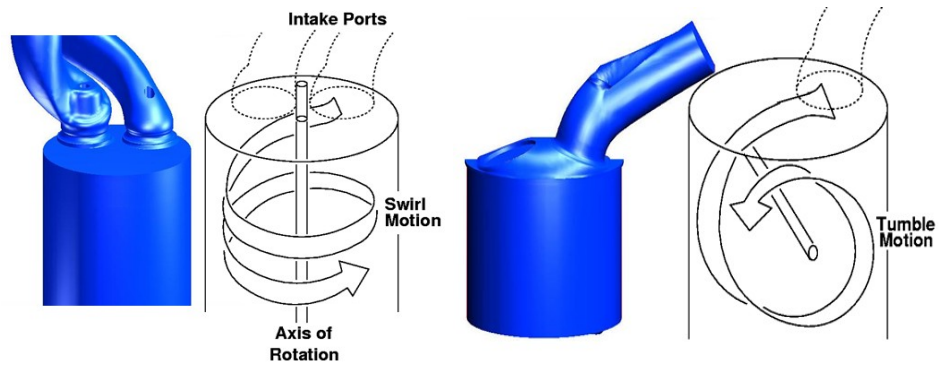

Figure 5: Investigating swirl and tumble flow with a comparison of flow visualization techniques. Image courtesy of Laramee et al. [4].

Computational fluid dynamics is the engineering discipline of tryingto predict fluid flow behavior: fluid motion as it interacts with geometries like cars, ships, or airplanes [4,5]. If we want to understand how fluid will interact with a surface one way to do, is to build an actual surface and build a flow environment and to visualize the flow with smoke or die or other substances. This is something fluid engineers do. This is a field of engineering. But it is very expensive. Just try a test flight and then attempt to visualize the air flow around the wings with smoke. This is a very expensive experiment. There are analog solutions. Can we come up with a digital solution that makes this investigation more feasible to accelerate the engineering and make it less expensive? That is the inspiration behind computational fluid dynamics (CFD).

Here's an example of combustion chamber in an automobile engine (Figure 5). The engineer's goal is to obtain a perfect mixture of fuel-to- air. The way the engineers propose to do that is to create this helical motion inside the combustion chamber (Figure 5 left) and for the diesel engine example the ideal pattern for the mixture of fluid flow is a tumble motion about an imaginary axis pointing out from the page (Figure 5 right).

We can build a real physical solution, but it saves time and money to go through a digital process first before we build real solutions. We don't need to build as many real solutions. The digital solution is computational fluid dynamics. As we know in computational fluid dynamics, the number one challenge is the amount of data that simulations generate which is at the gigabyte and terabyte scale. CFD simulations run from weeks to even months even on high performance computing machines. How can we use visualization to make sense of this massive amount of CFD data?

Figure 6: Pathlets visualizing tumble motion of flow. Image courtesy of Garth et al. [5]

Let's look at some data visualization solutions for CFD data, visualizing the swirl and tumble motion. See Figure 6. This is the tumble motion example so those are called path or short path lines in the flow direction and the color is mapped to crank angle. We have a piston head moving up-and-down a thousand cycles per minutes (at the bottom--not shown).

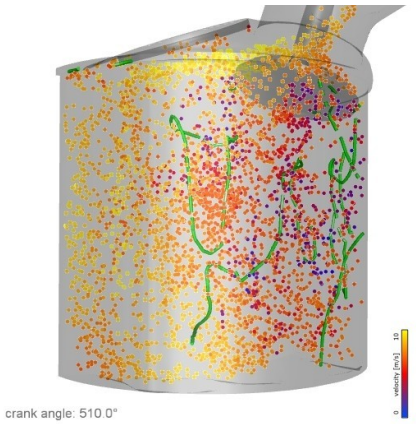

Figure 7: A hybrid visualization of particles and vortex core lines. Particles swirling around vortex cores aid the visualization of vortex core strength. Image courtesy of Garth et al.[4,5]

We can also use vortex corelines a combination of these green paths of vortex core lines: centers of swirling flow. See Figure 7. When combined with particles, the particles show flow behavior around the vortex cores. This is what we call feature-

based flow visualization -- looking for special features in the flow [4,5]. We can visualize the flow at the surface itself using so-called critical points. That's a sink and that's a saddle point. And then there are some curves that connect the different points. Those are special kind of streamline called separatrices and they show the topology of flow.

Topology is a skeletal representation of the flow. We can see the time-dependent topology of the flow-sinks, the sources, and the vortex corelines. The vortex core lines are tubes in the middle. We also see a separatrix on the boundaries of the surface and they are animated over time. That's a slow-motion version of time. It is slowed down quite a lot. In reality this is inside of the engine and it's moving up and down hundreds of times per minute. We can also use a volume visualization of the fluid flow specifically for the vortices so the vortices are the areas of swirling motion. The red is mapped to one direction of circular flow and blue the other.

The idea is to visualize the swirl and tumble motion. In this case the tumble motion is about an imaginary axis that points out at the at the viewer just like a tumble dryer. We can observe an axis pointing out and downwards to the left of the ideal axis. It's a very unstable axis of rotation. That is what these visualizations show an unstable rotational axis.

The computation fluid dynamicists see this and they observe this is not the ideal tumble motion. There's a little bit of tumble motion right around the perimeter of the geometry but as soon as we look in the center we still see some swirling motion but it's very far from the ideal kind of tumble motion they strive for. They have to make some modifications to the geometry to try to realize the best mixing possible. And here this is also not the ideal swirl motion. The motion is off-center. Again they have not achieved their target of the ideal motion. That's what these visualizations show. They show the difference between the actual and predicted motion.

One of the things that the engineers like to know is where precisely the flow is misbehaving. They know what they want to see and what they expect to see. They like to see visualizations that highlight unwanted behavior. That's what all users want to see. In fact, that could be in the knowledge evolution pipeline. One of the things QPC would like to see where the abandoned calls are and when people are not behaving properly. Here the engineers can see where the flow does not behave properly. This is one of the strengths of visualization -- to show when and where behavior go wrong.

7. Example: Visualization of Sensor Data from Animal Movement

The next example is from marine biology. Marine biologists would like to understand marine wildlife and how marine wild life behaves. One of the challenges that they face is deep sea underwater diving. How do you study animals that dive deep underwater for hours or even days at a time? How is that possible? Theoretically the solution might be to follow the animal. That might be kind of an approach. But there are some problems with that. People cannot just dive a few kilometers underneath the water. They can try to build submarines or similar but to try to follow a cormorant or tortoise in a submarine is not a very practical solution. It's not feasible, very expensive, and the analog solution is one of those cases where the observation itself influences the behavior we are trying to study.

Marine biologists look to the digital world for a solution. They use sensor devices at Swansea University called a daily diary [6]. They actually capture the animals like a cormorant. They attach the digital sensor or maybe more than one digital sensor to the subject and then release it. See Figure 8.

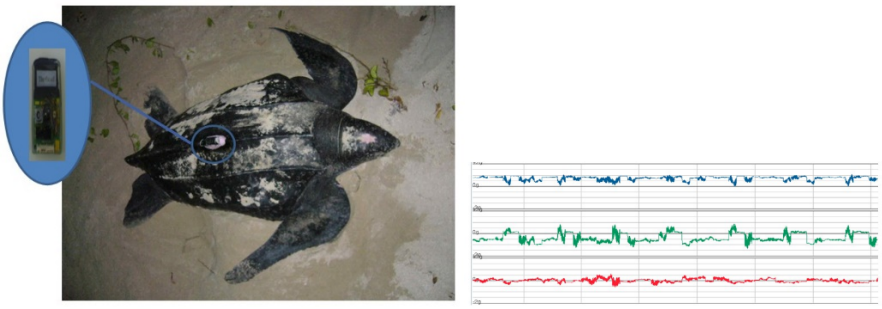

Figure 8: Visualization of Sensor Data from Animal Movement. Image courtesy of Grundy et al. [6]

Then they recapture the sensor a few hours or a few days later. They remove it from the animal and they study the information that it collects about the local environment. It collects information on acceleration, local acceleration, local temperature, pressure, ultraviolet light, and a few other properties. Another challenge currently is that GPS does not work underwater at great depths. It's not possible to just plot a path naively in a dead reckoning fashion the same way we can for land animals.

However, when the user get this data this is what it looks like (See Figure 8 right). This is a tiny little piece of what it looks like. They plot, for every attribute, magnitude versus time. Acceleration Magnitude is on the y-axis and time is on the x-axis. They claim they can infer animal behavior based on these wave patterns. They can look at a wave pattern and say that it looks like the animal is diving or the animal hunting.

But you can see that that's not easy. This is only a few seconds of data. If you plot the day's worth of data in this fashion, it will wrap around a building a few times. The acceleration has three components: *x, y, z*. These are three components decoupled. In reality they form a vector in 3-space.

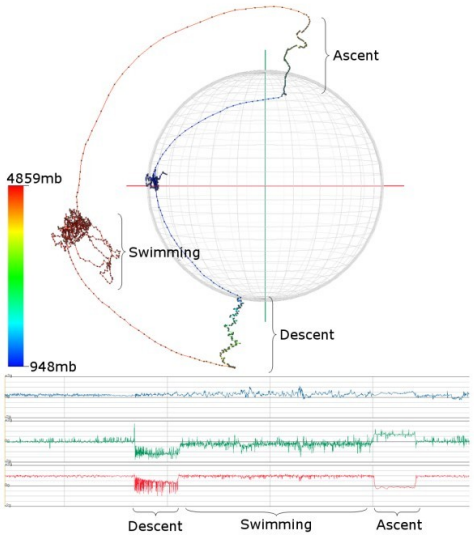

Figure 9: Spherical visualization of sensor data coupled with standard visualization (bottom). Image courtesy of Grundy et al.[6]

The marine biologists asked us if we can drive visualizations that facilitate the understanding of marine wildlife behavior. We have a standard visualization coupled with a new visualization. (See Figure 9.) In the new visual design we can see the geometry of the animals and how the animal is oriented immediately. What Grundy

et al. did was reintegrate the *x, y, z* components of the acceleration and plot them in spherical space rather than time versus amplitude space. And they map the unit vectors onto a sphere and can immediately infer animal behavior. They can also map pressure to the radius. Figure 9 shows the animal swimming at the surface and then the pressure increases. Pressure mapped to radius represents diving behavior and the diving behavior is very easy to notice. Now that is visualized in spherical space we can observe swimming, hunting, searching behavior. This spherical space is interactive so that we can rotate, zoom, and pan at different angles.

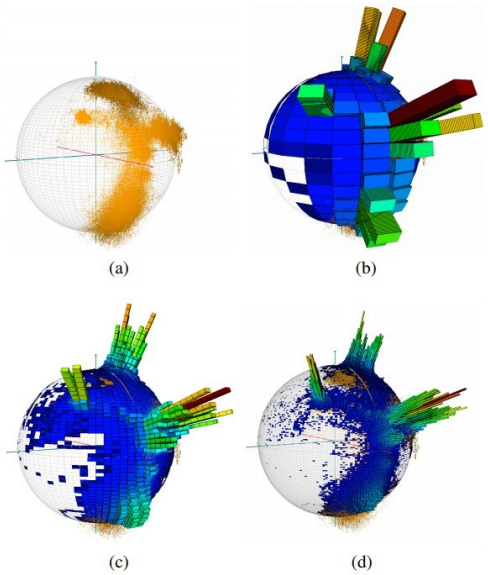

Figure 10: Spherical histogram of sensor data. Image courtesy of Grundy et al.[6]

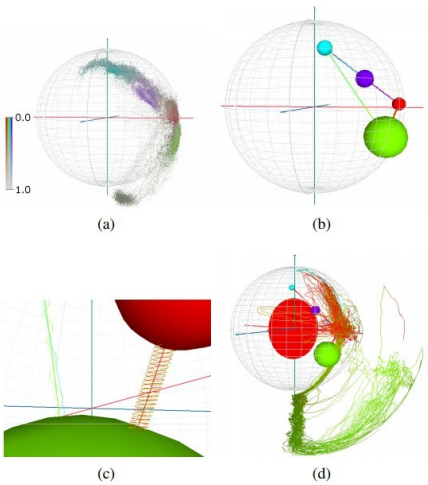

Figure 11: Utilizing data clustering methods of sensor data. Image courtesy of Grundy et al.[6]

Figure 10 presents a spherical histogram. The vectors are binned into unit rectangles and the more time an animal spends in a given posture at that orientation, the longer histogram bin. We can see the postures and the states that the animals spend a long time in. Rather than focusing on all of the time, the user chooses a special region and then the region is plotted up close in the left-hand corner. The user can cluster the vectors into different groups. (See Figure 11). Assigning each data point to

a group that represents some interesting aspect of the animal behavior. The user can adjust the probability of any data sample belonging to one of the clusters. These are clusters of animal postures calculated using *K*-means clustering. Grundy et al. can represent clusters as spheres and then connect the spheres or the postures with edges that represent transitions from one orientation to another successively. We can observe the transitions between various states and postures. We can see the most popular or dominant states. That information pops up immediately.

8. Example: Visualization of Molecular Dynamics Simulation Data

The goal here is to understand biology at the molecular level. There are analog approaches and solutions to this challenge. Biologists run experiments at the molecular level and try to understand behavior of molecules using experiments and nuclear magnetic resonance spectroscopy. These machines and experiments are very expensive.

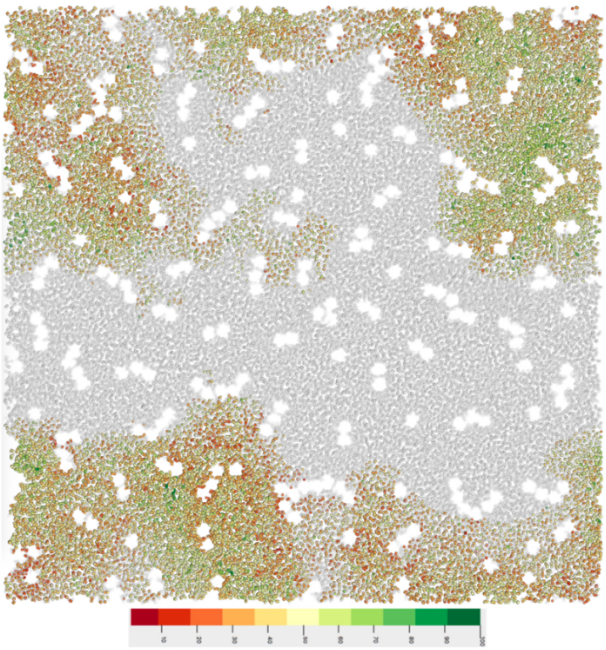

Figure 12: Visualization of molecular dynamics simulation data. Image courtesy of Alharbi et al. [7]

The whole field of computational biology attempts to address this challenge in the digital world because it's much less expensive than the analog world. As with any simulation data all the simulation experts generate massive amounts of data. They try to use the latest high-performance computing machines.

This is the interaction of lipids and proteins. See Figure 12. That's what this simulation data shows and Alharbi et al. [7] develop some visualization software to enhance understanding of this. These holes are protein and then the paths are lipid trajectories. See Figure 12. The computational biologists attempt to visualize the interaction between trajectories and the proteins.

Alharibi et al. are trying to develop visualizations to help computational biologist understand the data with a special focus, in this case, on path filtering. Given the massive number of trajectories hundreds of thousands or millions of trajectories over multiple time steps, is it possible to select a subset of those trajectories based on

interesting properties that help the biologists understanding the behavior? Alharibi et al. develop tools for filtering and selection of these trajectories to try to understand behavior. One example is just changing the time step of the simulation or filtering the path by its length. They can focus on shorter paths or on longer paths. They can slide the filter over to long paths or the long trajectories.

The user can filter the paths based on other characteristics. They chose a few properties that they hope will be interesting for the computational biologists. One property is curvature. There are highly curved paths.

The atom trajectories are actually three dimensions, but they're limited to a layer analogous to the biosphere such that the *z* dimension is relatively small compared to the *x* and *y* dimensions. They can visualize projected 2D space or the volumetric 3-space. The user can experiment with 2D versus 3D. The standard visualization packages for this are constrained to a two-dimensional plane and they're generally not interactive.

Conclusion

This chapter presents a ubiquitous model of knowledge evolution witnessed at a collective level by a society deeply involved with the digital world. It presents a theory supported by a number of case studies ranging from the call center industry, to automotive engineering, to computational biology. It sets the stage for data visualization as a vital technology to evolve our understanding of data and the world it describes to the next level. It will be exciting to witness how this model and pattern evolve over time.

References

[1] Data Visualization definition. Available from: https://searchbusinessanalytics.techtarget.com/definition/data-visualization [Accessed: 2018-06]

[2] C. Ware, Information visualization: perception for design. Elsevier, 2012.

[3] R. Roberts, C. Tong, R. Laramee, G. A. Smith, P. Brookes, and T. D'Cruze, "Interactive Analytical Treemaps for Visualization of Call Centre Data," in Proceedings of the Conference on Smart Tools and Applications in Computer Graphics. Eurographics Association, 2016, pp. 109–117.

[4] R. S. Laramee, J. Schneider, and H. Hauser, "Texture-based flow visualization on isosurfaces." in VisSym, 2004, pp. 85–90.

[5] C. Garth, R. S. Laramee, X. Tricoche, J. Schneider, and H. Hagen, "Extraction and visualization of swirl and tumble motion from engine simulation data," in Topologybased Methods in Visualization. Springer, 2007, pp. 121–135.

[6] E.Grundy, M.W.Jones, R.S.Laramee, R.P.Wilson, andE.L.Shepard, "Visualization of sensor data from animal movement," in Computer Graphics Forum, vol. 28, no. 3. Wiley Online Library, 2009, pp. 815–822

[7] N. Alharbi, R. S. Laramee, and M. Chavent, "Molpathfinder: interactive multidimensional path filtering of molecular dynamics simulation data," in The Computer Graphics and Visual Computing (CGVC) Conference, vol. 2016, 2016, pp. 9–16.